

Value Drivers in a Changing Landscape of Modeling & Simulation

Pieter J. Mosterman

Chief Research Scientist, Director
Advanced Research & Technology Office (MARTO)



Adjunct Professor
School of Computer Science





\$1,052

Get Your Blue Book® Value then Price Your Next Car

1992 Mitsubishi Galant LS Sedan 4D | Mileage: 210,000

Tell us your car's options or [See value with standard equipment](#)

Standard equipment pre-selected below

Engine

☒ 4-Cyl, 2.0 Liter

Drivetrain

☒ FWD

Comfort and Convenience

- ☒ Air Conditioning
- ☒ Power Windows
- ☒ Power Door Locks
- ☒ Cruise Control

Seats

☐ Leather

Roof and Glass

- ☐ Sun Roof (Flip-Up)
- ☐ Sun Roof (Sliding)
- ☐ Moon Roof

Transmission

- ☒ Automatic
- ☐ Manual, 5-Spd

Steering

- ☒ Power Steering
- ☒ Tilt Wheel

Entertainment and Instrumentation

- ☒ AM/FM Stereo
- ☒ Cassette
- ☐ CD (Single Disc)
- ☐ CD (Multi Disc)

Wheels and Tires

- ☒ Steel Wheels
- ☐ Alloy Wheels
- ☐ Premium Wheels

Search Local Listings

- ☒ Search Mitsubishi Galant
- ☐ Search All Cars for Sale near Framingham

[Search](#)

[More Mitsubishi Galant Vehicles for Sale](#)



\$11,221

Moore's Law

Get Your Blue Book® Value then Price Your Next Car

2012 Mitsubishi Galant SE Sedan 4D | Mileage: 50,000

Tell us your car's options or [See value with standard equipment](#)

Standard equipment pre-selected below

Engine

☒ 4-Cyl, 2.4 Liter

Transmission

☒ Auto, 4-Spd w/Sportronic

Drivetrain

☒ FWD

Braking and Traction

- ☒ Traction Control
- ☒ Active Stability Control
- ☒ ABS (4-Wheel)

Comfort and Convenience

- ☒ Keyless Entry
- ☐ Keyless Start
- ☒ Air Conditioning
- ☒ Power Windows
- ☒ Power Door Locks
- ☒ Cruise Control

Steering

- ☒ Power Steering
- ☒ Tilt Wheel

Safety and Security

- ☒ Backup Camera
- ☒ Dual Air Bags
- ☒ Side Air Bags
- ☒ F&R Head Curtain Air Bags

Entertainment and Instrumentation

- ☒ AM/FM Stereo
- ☒ MP3 (Multi Disc)
- ☒ Rockford Premium Sound
- ☒ SiriusXM Satellite
- ☒ Navigation System
- ☒ Bluetooth Wireless
- ☒ Multi-Communication Sys

Seats

- ☒ Heated Seats
- ☒ Power Seat
- ☐ Leather

Roof and Glass

- ☒ Moon Roof

Search Local Listings

- ☒ Search Mitsubishi Galant
- ☐ Search All Cars for Sale near Framingham

[Search](#)

[More Mitsubishi Galant Vehicles for Sale](#)



\$35,000

Moore's Law

Get Your Blue Book® Value then Price Your Next Car



“When one car learns something, the whole fleet learns it”

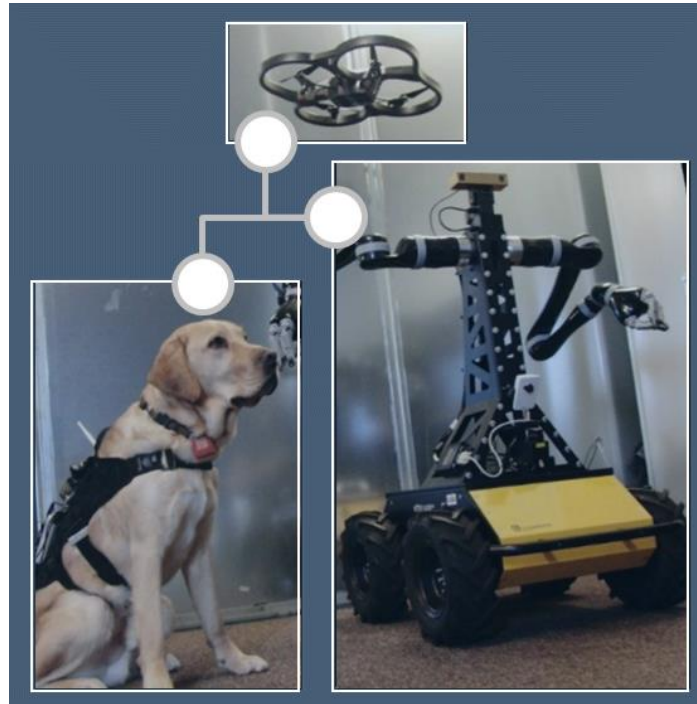
Metcalfe's Law



Elon Musk

Machines are **connecting**
and **collaborating**

Where can we have
impact, which **solutions**
are needed, what
challenges these
solutions, and how can we
overcome the challenges?



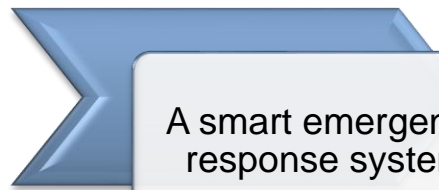
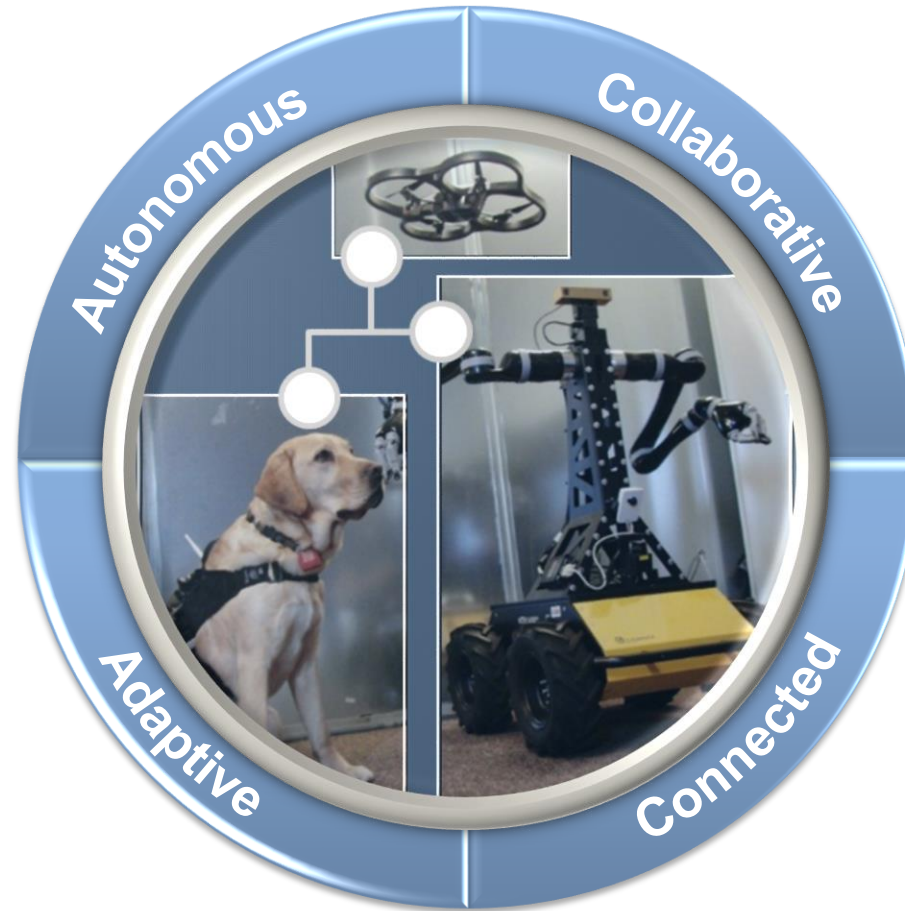
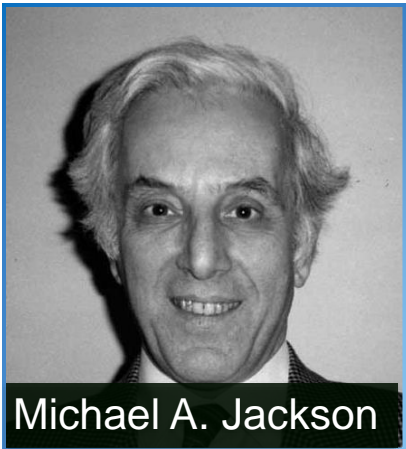
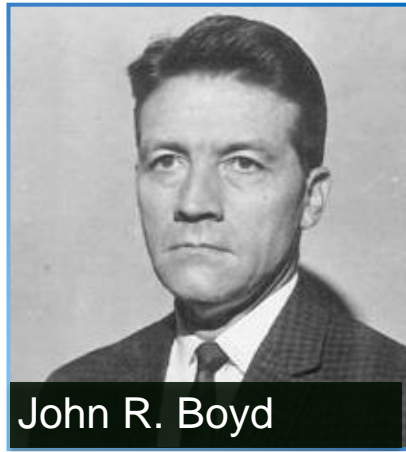
A smart emergency
response system

A smart emergency
response system

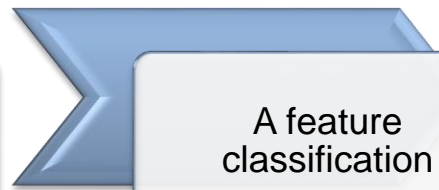
A feature
classification

Evolving
architectures

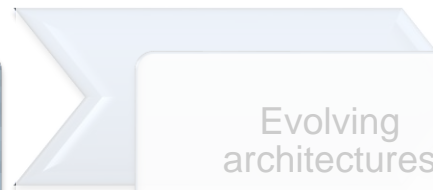
Future value drivers



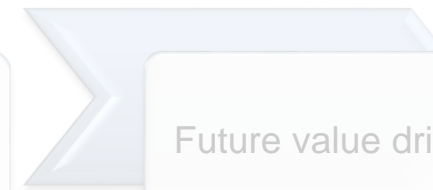
A smart emergency response system



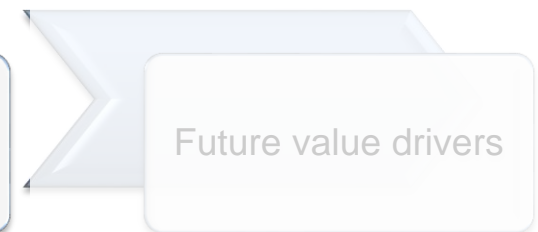
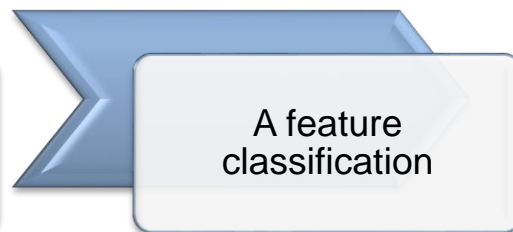
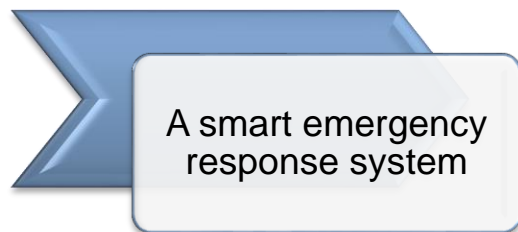
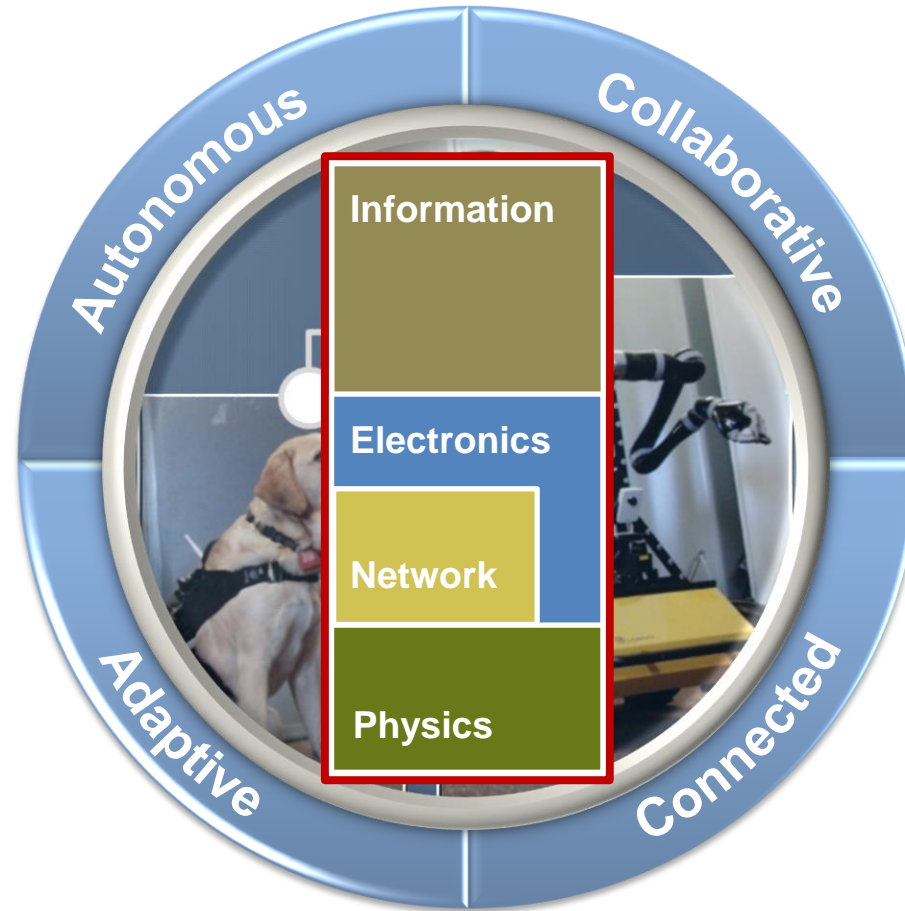
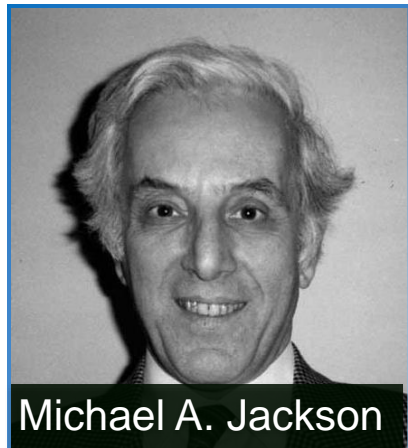
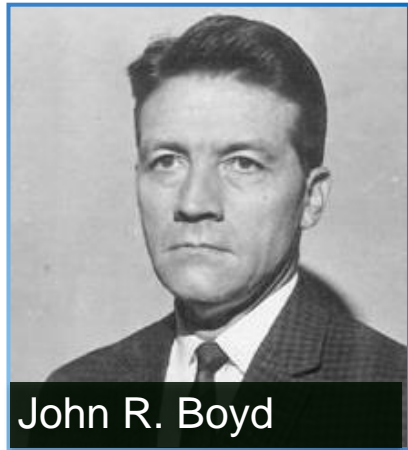
A feature classification

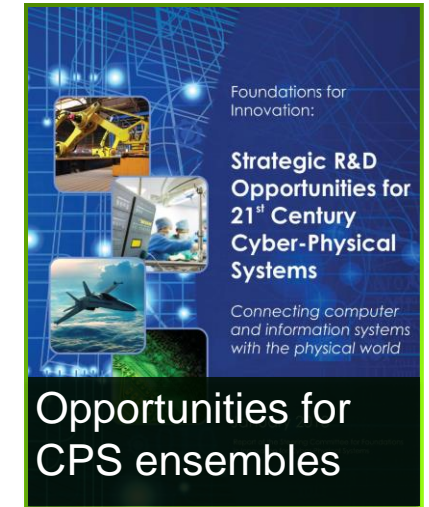
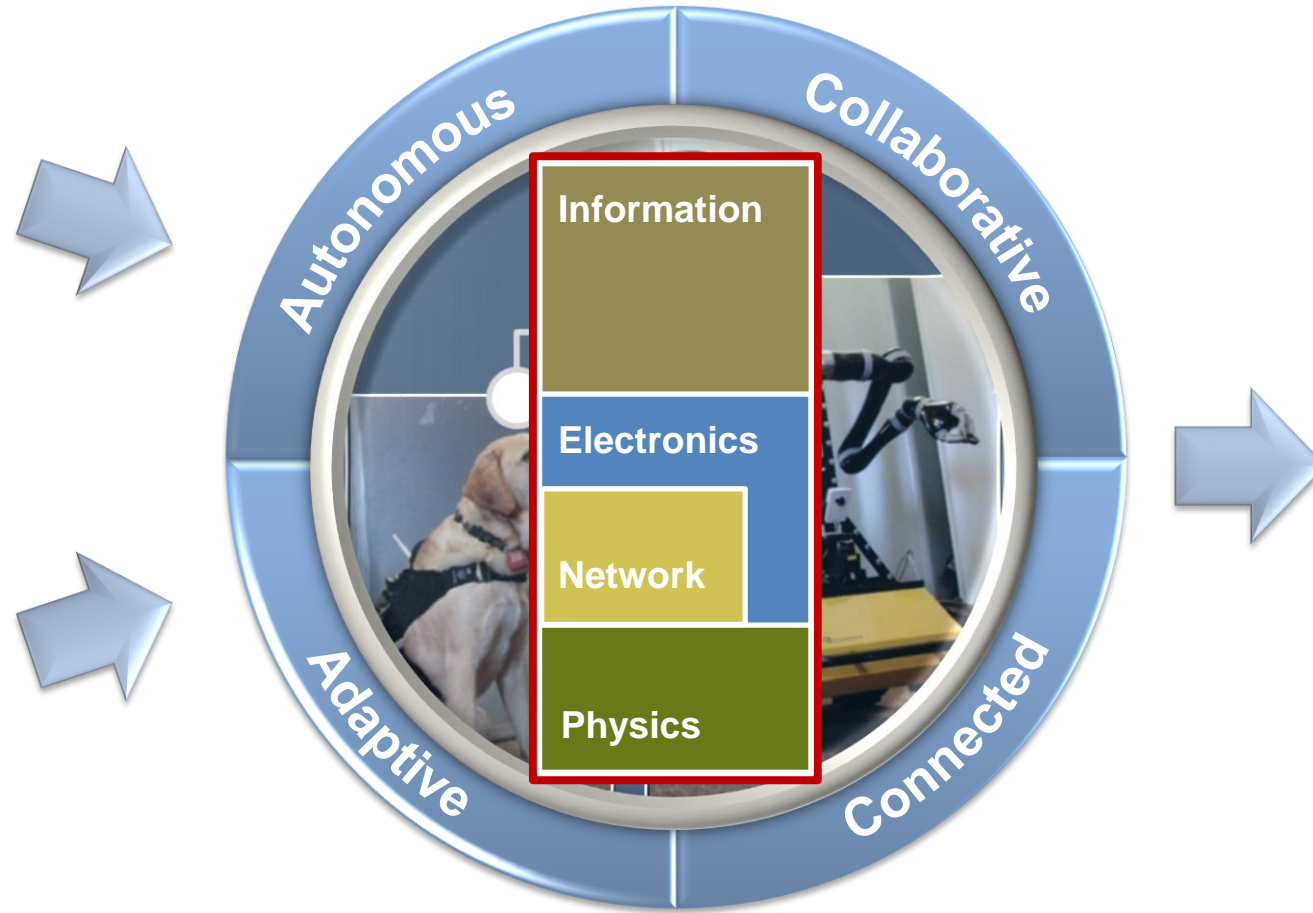
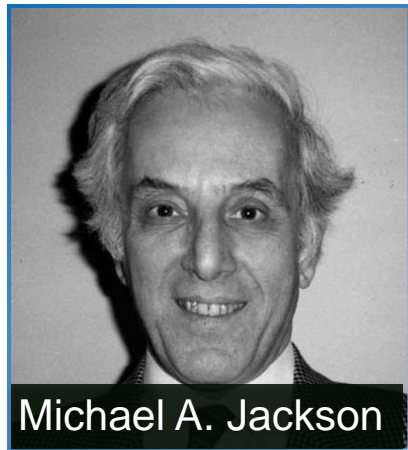
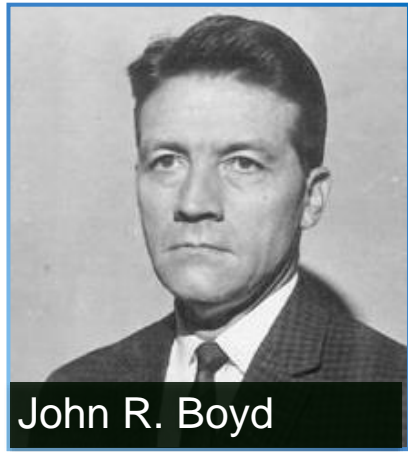


Evolving architectures



Future value drivers





2013 MathWorks Summer Research Internship



Kun Zhang
University of Arizona



Enes Bilgin
Boston University



David Escobar Sanabria
University of Minnesota

İzmit, Turkey, 1999







Where can we help make a difference?



Deploy a heterogeneous fleet ...



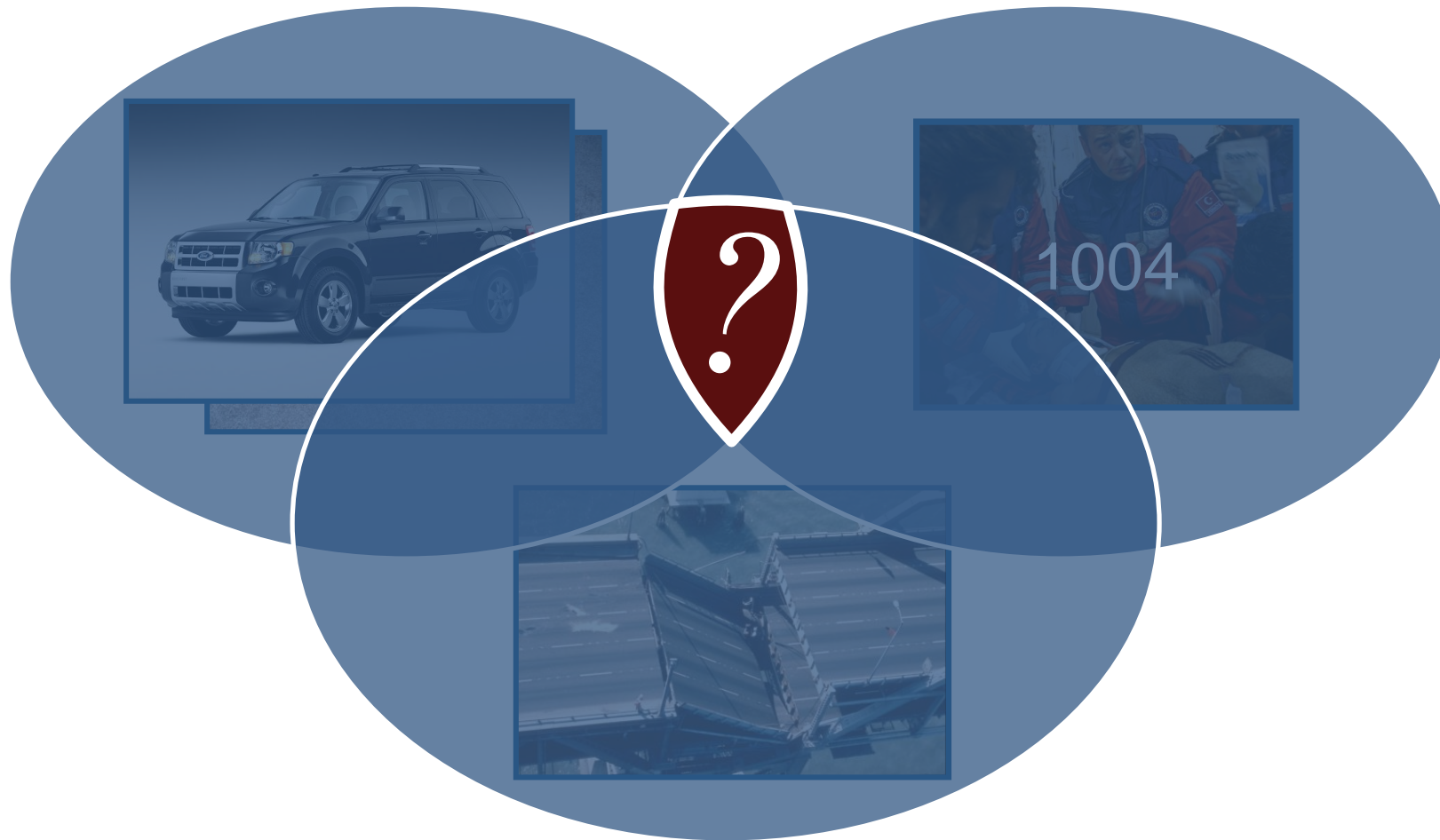
... to serve many (changing!) requests ...

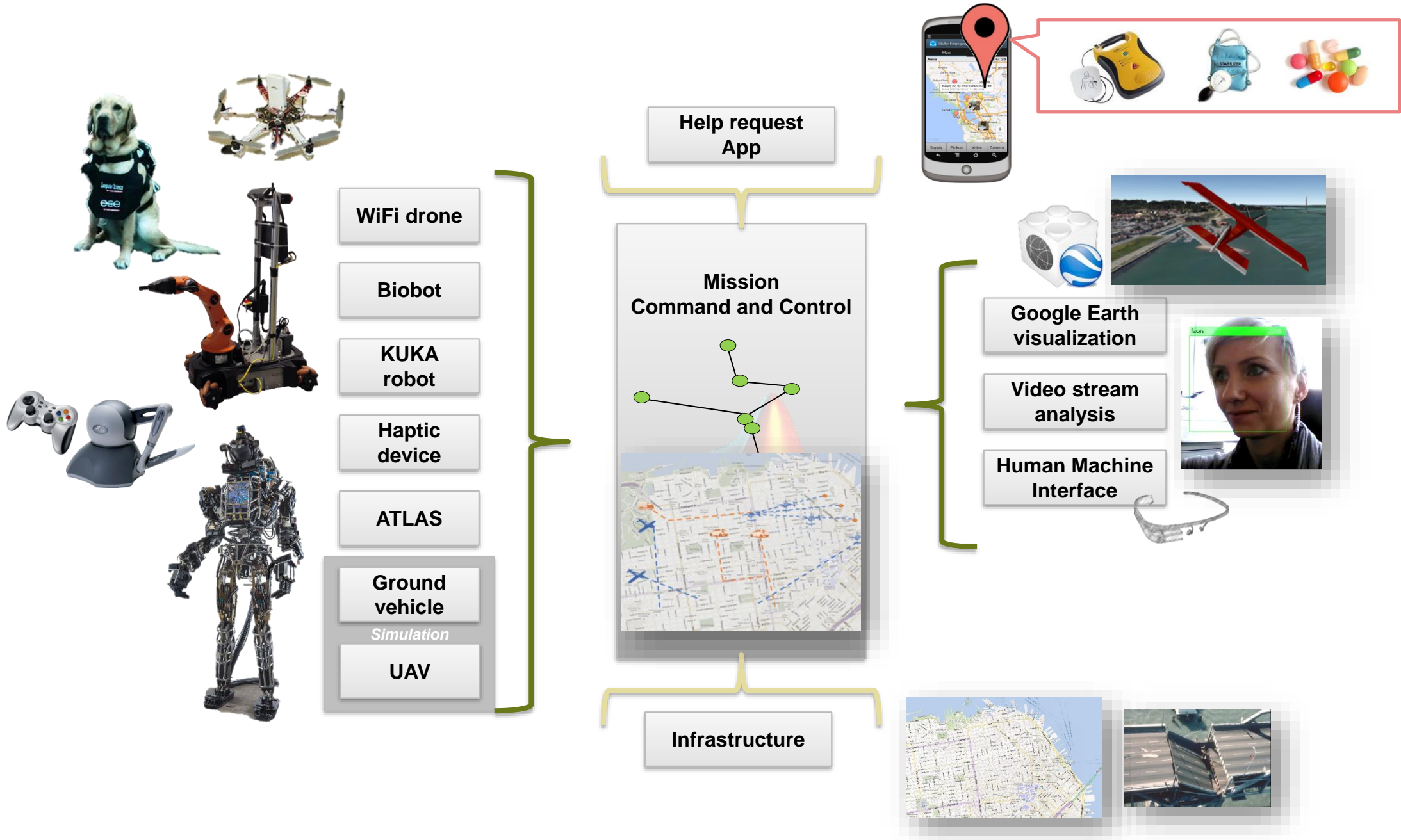


... across an uncertain infrastructure ...



... **all** in a time optimal manner!







2013 MathWorks Summer Research Internship: A Spectacular Challenge (Get cyber real!)

<https://youtu.be/MxrySx1m8VQ?t=2m42s>

Image Landsat

Google earth

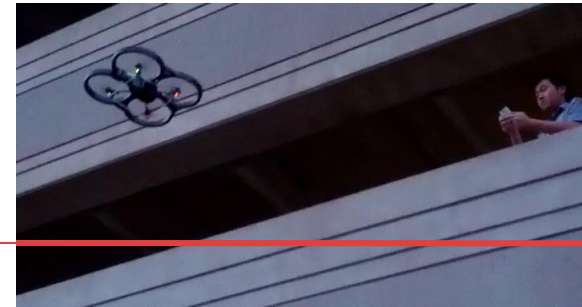
Man-machine control transfer



hover



transfer control



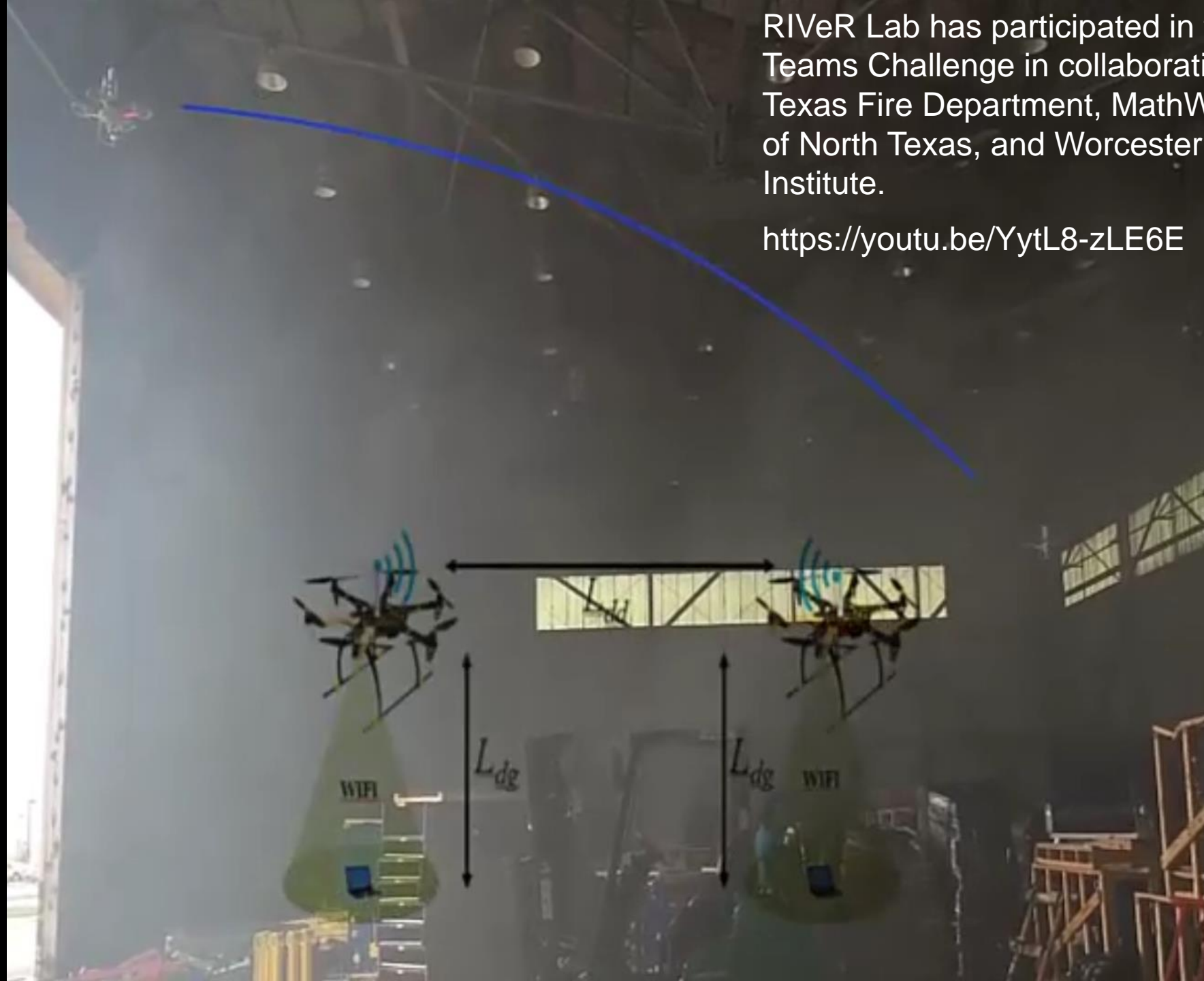
control by responder

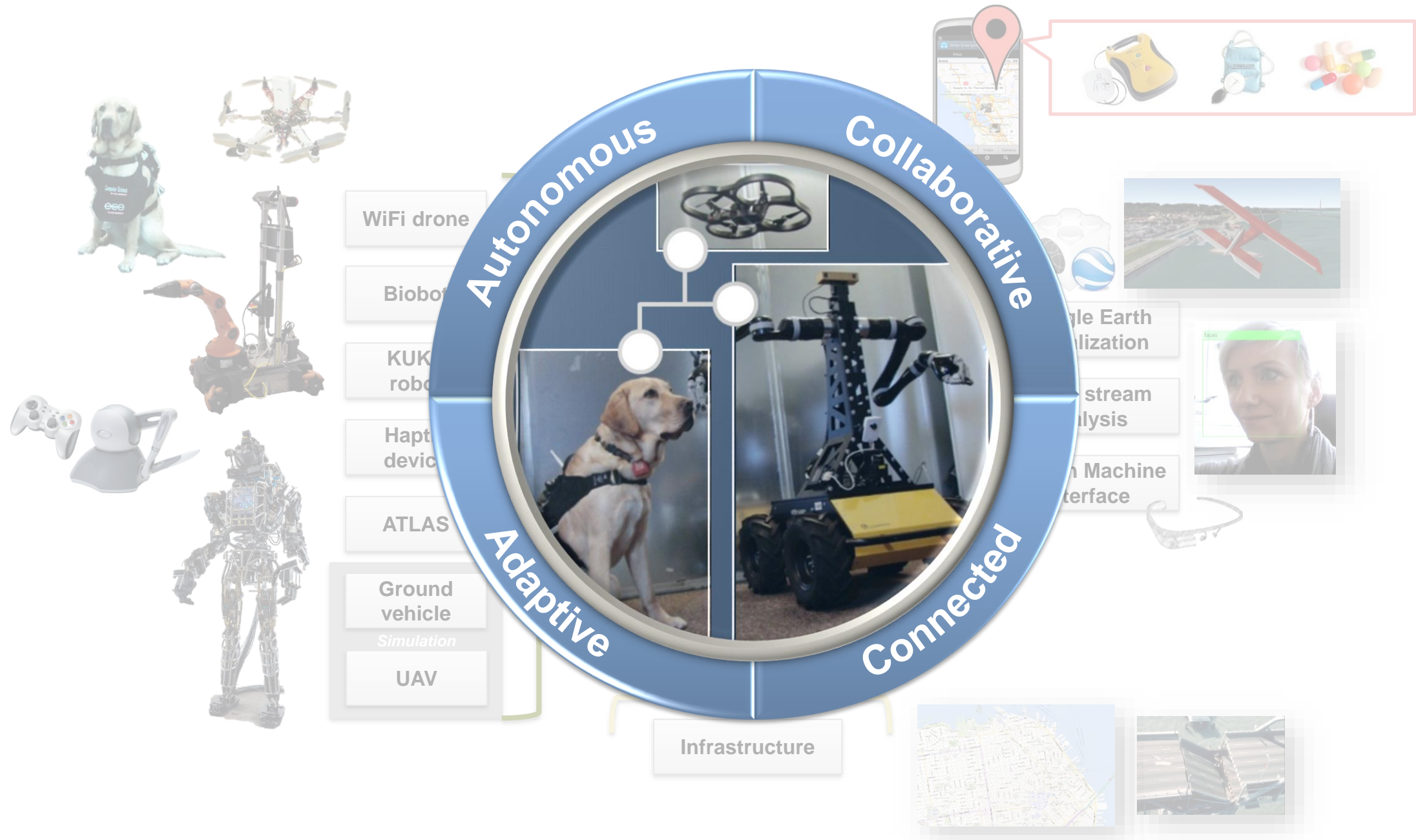
2013 MathWorks Summer Research Internship: A Self-flying Drone (Take cyber control!)

<https://youtu.be/M3vq1ywbe10?t=41s>

RIVeR Lab has participated in the Global City Teams Challenge in collaboration with Austin Texas Fire Department, MathWorks, University of North Texas, and Worcester Polytechnic Institute.

<https://youtu.be/YytL8-zLE6E>



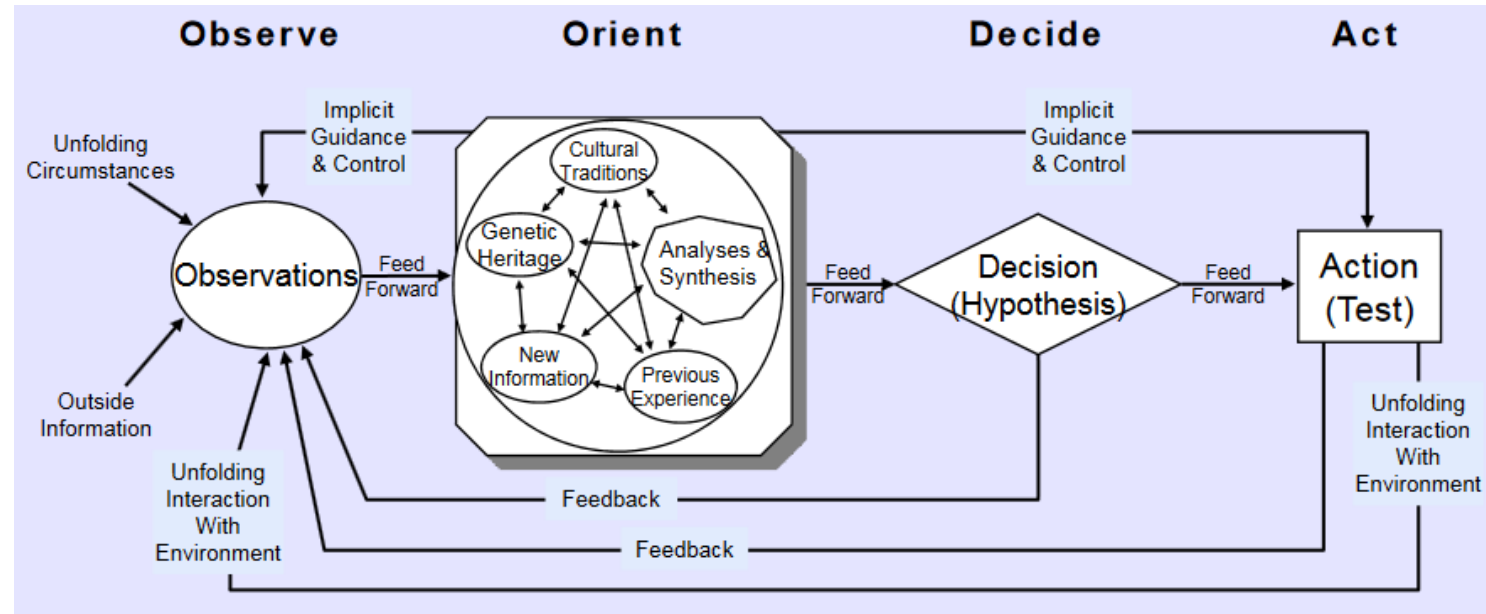




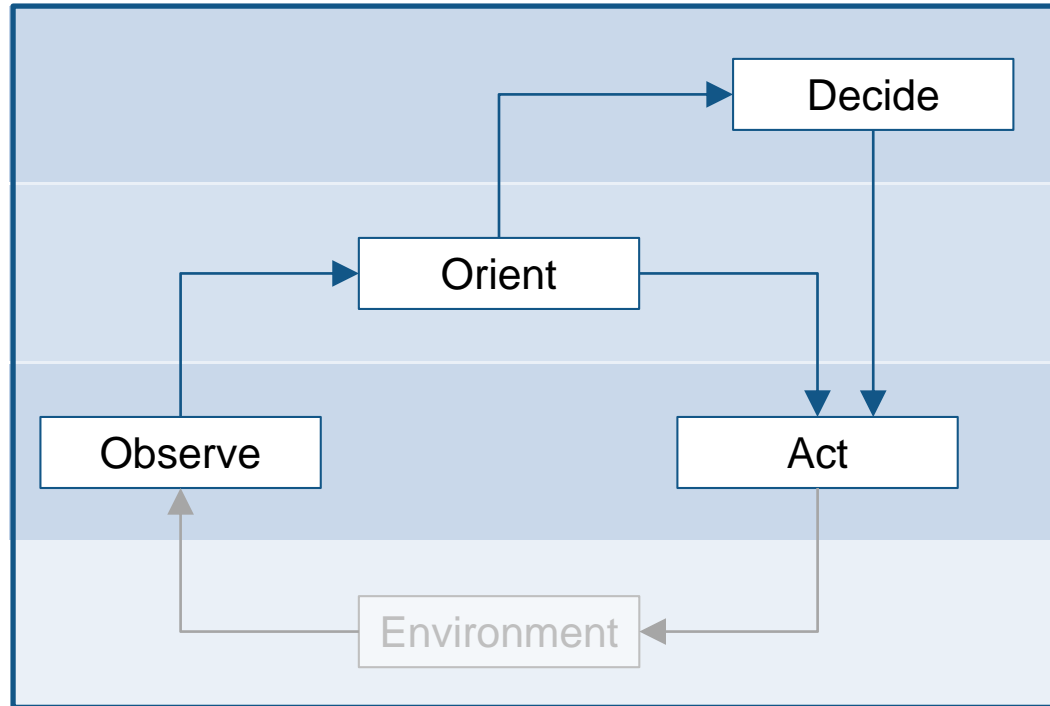
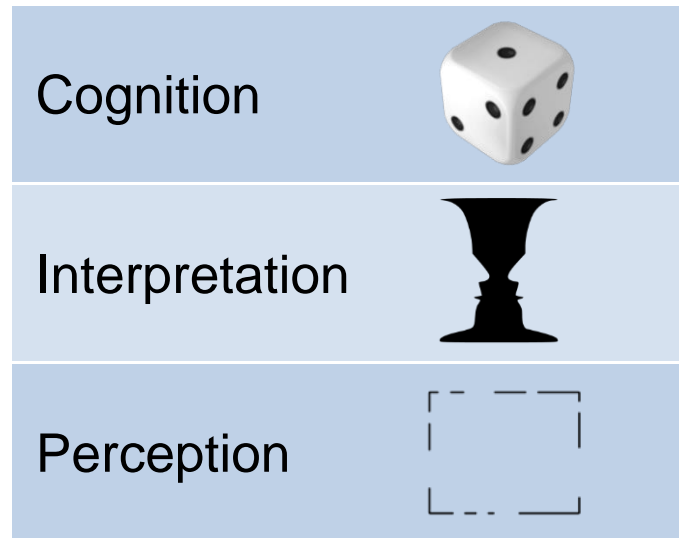


Colonel John Richard Boyd

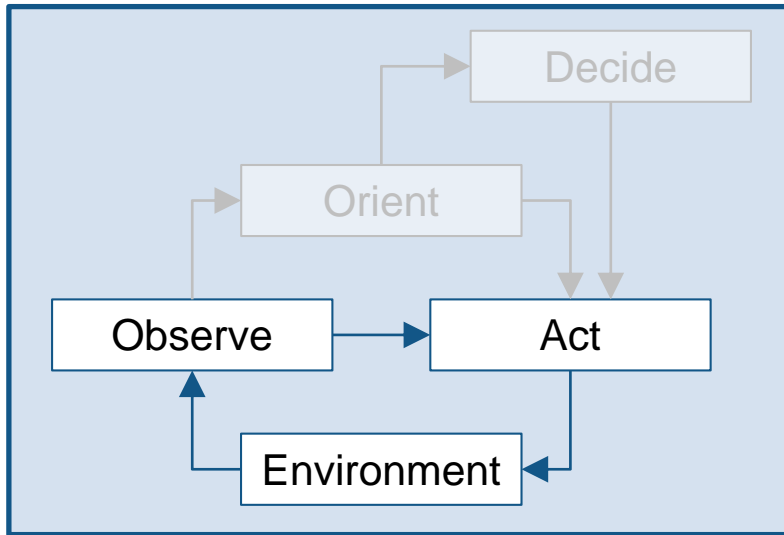
The Observe-Orient-Decide-Act (OODA) loop



OODA and the stages of cognition



Engineered systems and the stages of cognition

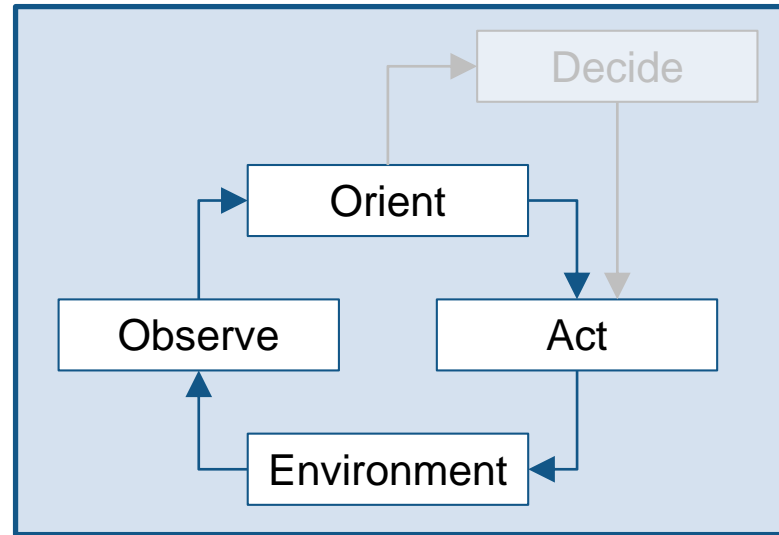


Automatic

Perception

Process (video)
Analyze for validity (filter, reject)

Compute control signals

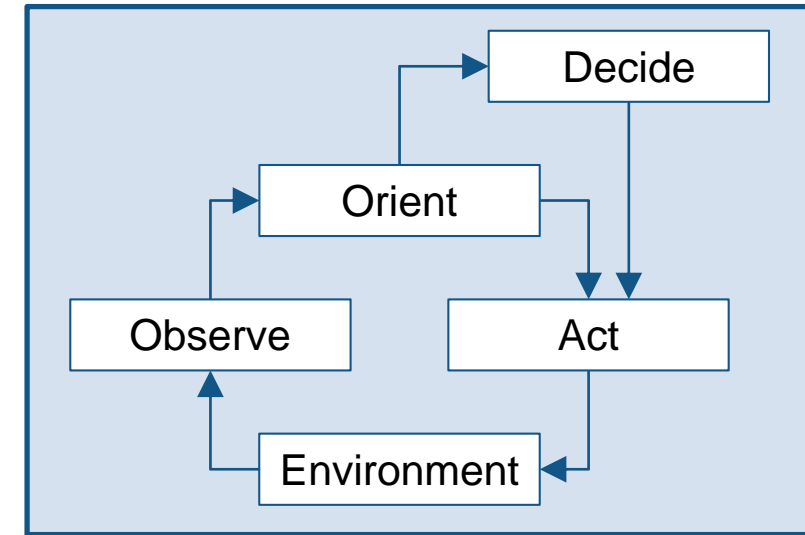


Adaptive

Interpretation

Map to semantic concepts
Fuse sensor data

Adjust control
Reconfigure behavior



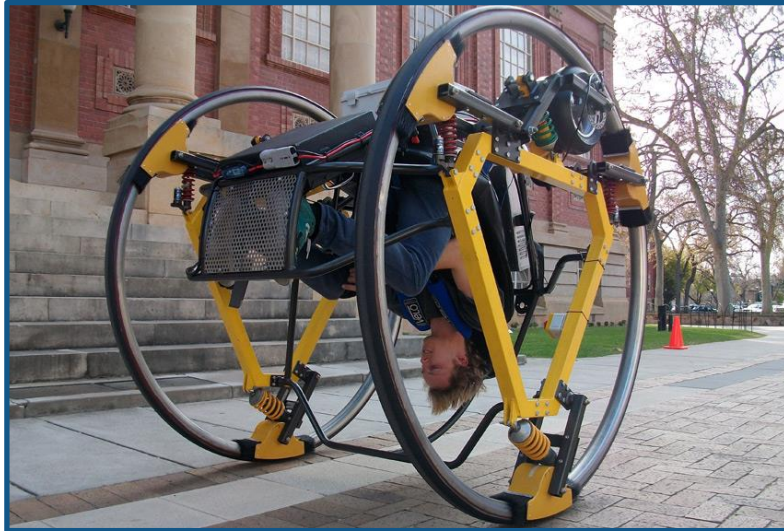
Autonomous

Cognition

Reason based on knowledge
Plan for objectives and constraints

Assess alternatives
Determine course of action

Engineered systems and the stages of cognition



Automatic

Perception

Process (video)
Analyze for validity (filter, reject)

Compute control signals



Adaptive

Interpretation

Map to semantic concepts
Fuse sensor data

Adjust control
Reconfigure behavior



Autonomous

Cognition

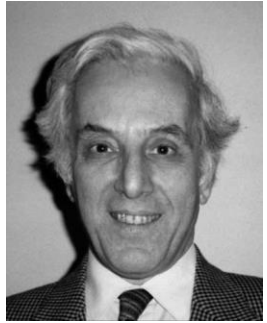
Reason based on knowledge
Plan for objectives and constraints

Assess alternatives
Determine course of action

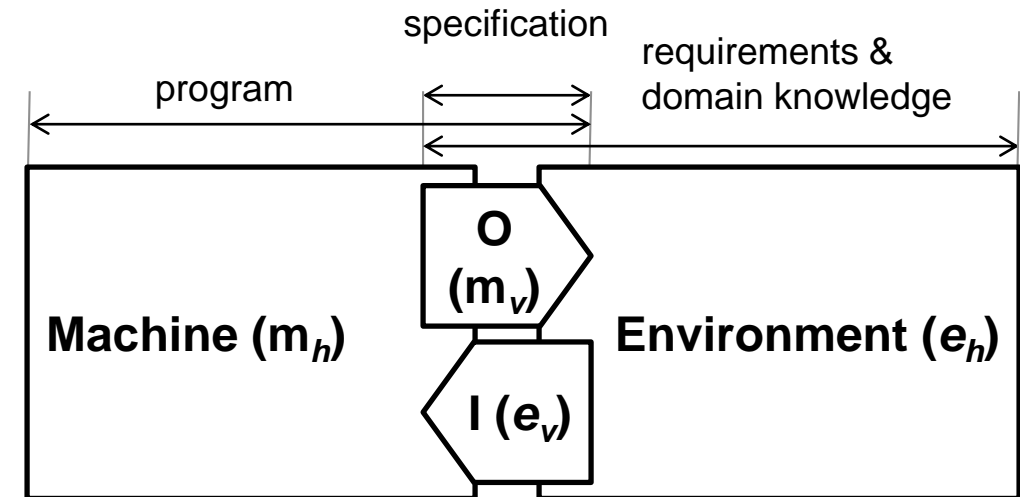


Michael Anthony Jackson

Requirements engineering



- A **requirement** is a desired relationship among the **phenomena** (e.g., actions/events, states) of the environment
- Phenomena are categorized as
 - e_h : controlled (or initiated) by the **e**nvironment and **h**idden from (i.e., invisible to, not shared with) the machine
 - e_v : controlled by the **e**nvironment but **v**isible to (i.e., shared with) the machine
 - m_v : controlled by the **m**achine but **v**isible to (shared with) the environment
 - m_h : controlled by the **m**achine and **h**idden from (i.e., not shared with) the environment



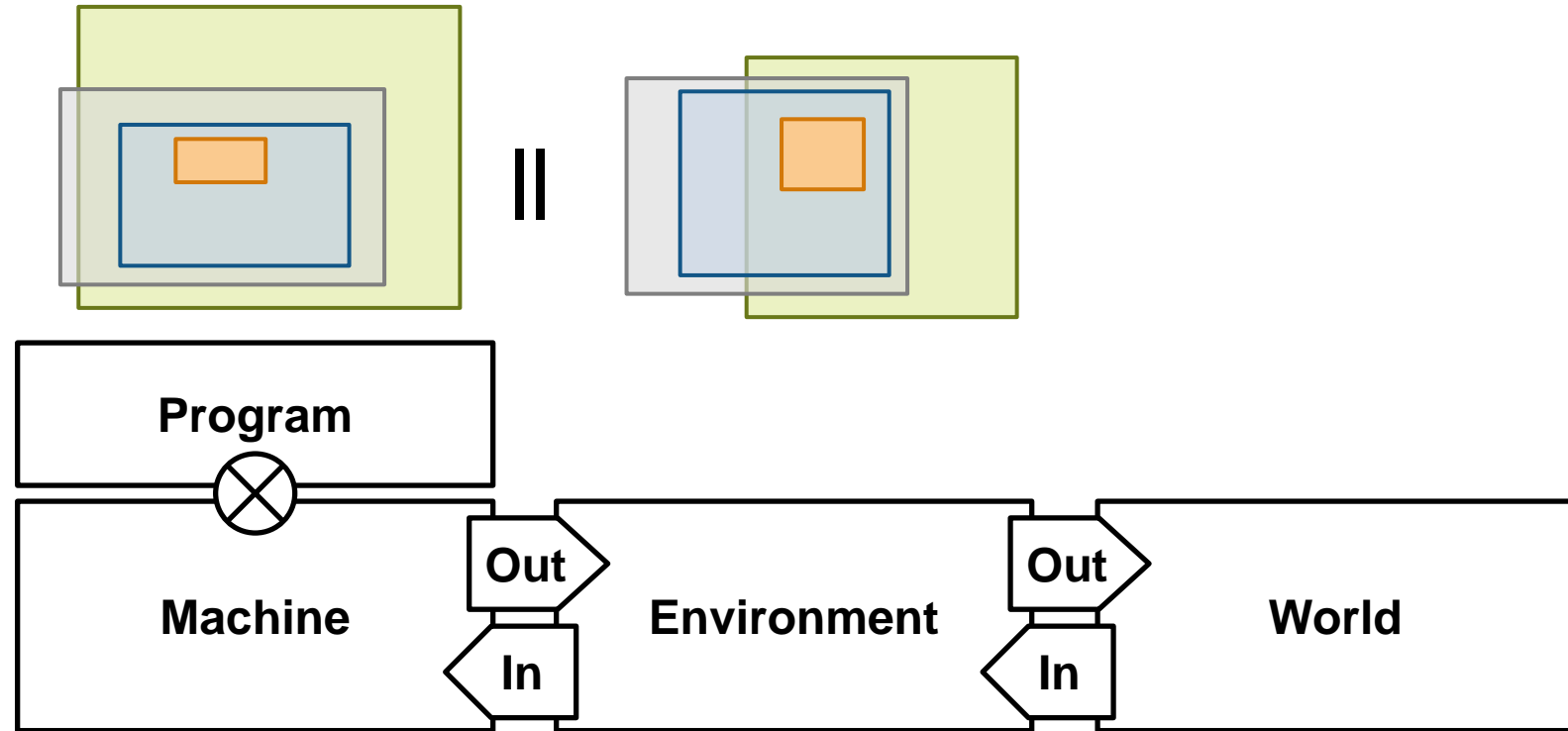
A behavioral view

Closed loop designed behavior

Property satisfying behavior

Closed loop possible behavior

Open loop possible behavior



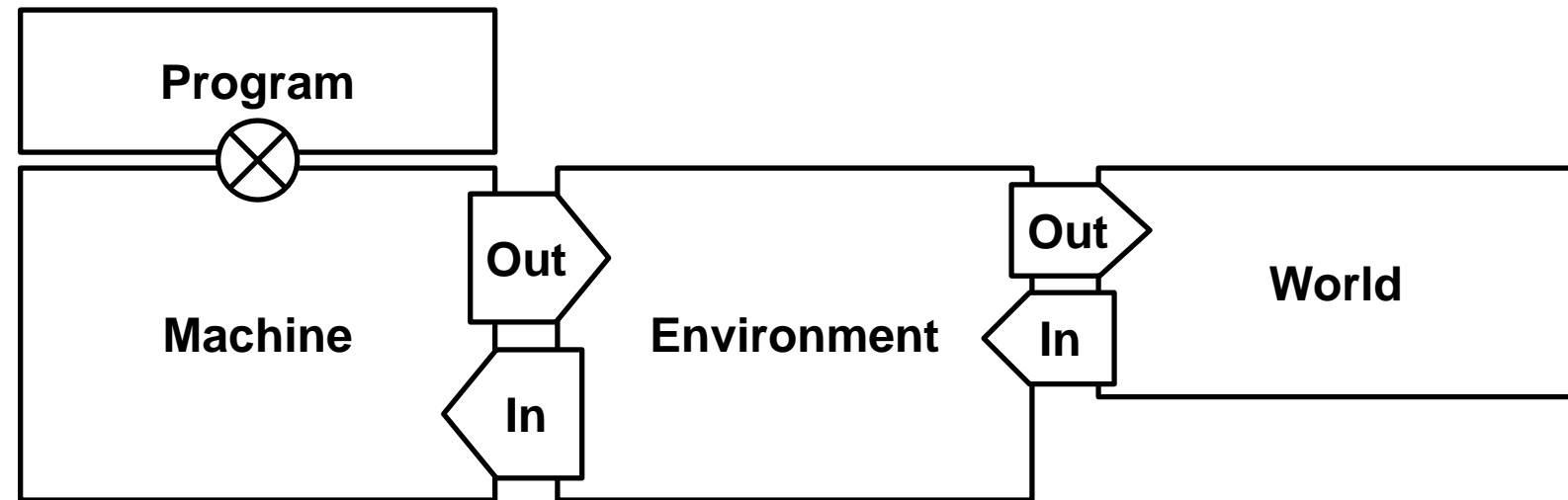
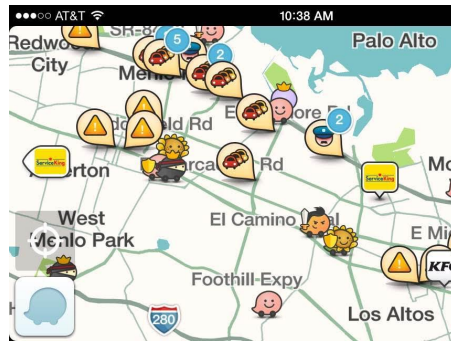
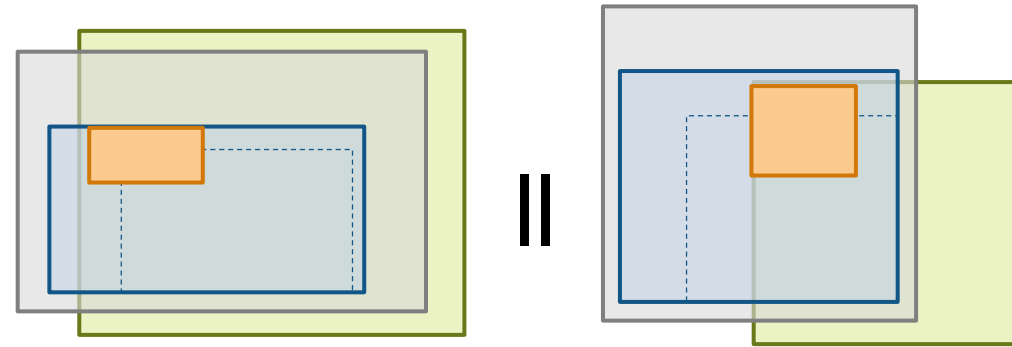
A behavioral view

Closed loop designed behavior

Property satisfying behavior

Closed loop possible behavior

Open loop possible behavior



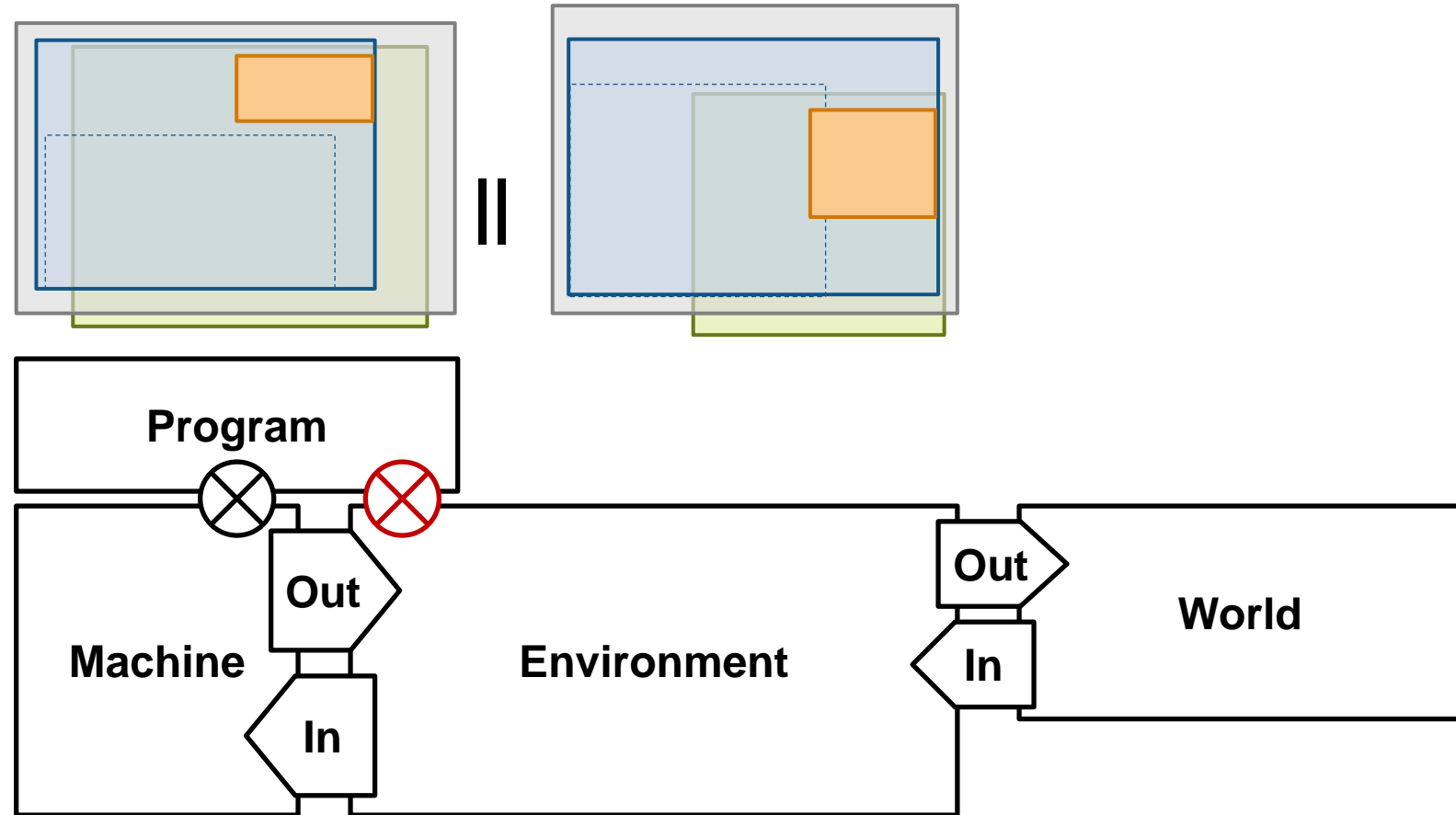
A behavioral view

Closed loop designed behavior

Property satisfying behavior

Closed loop possible behavior

Open loop possible behavior

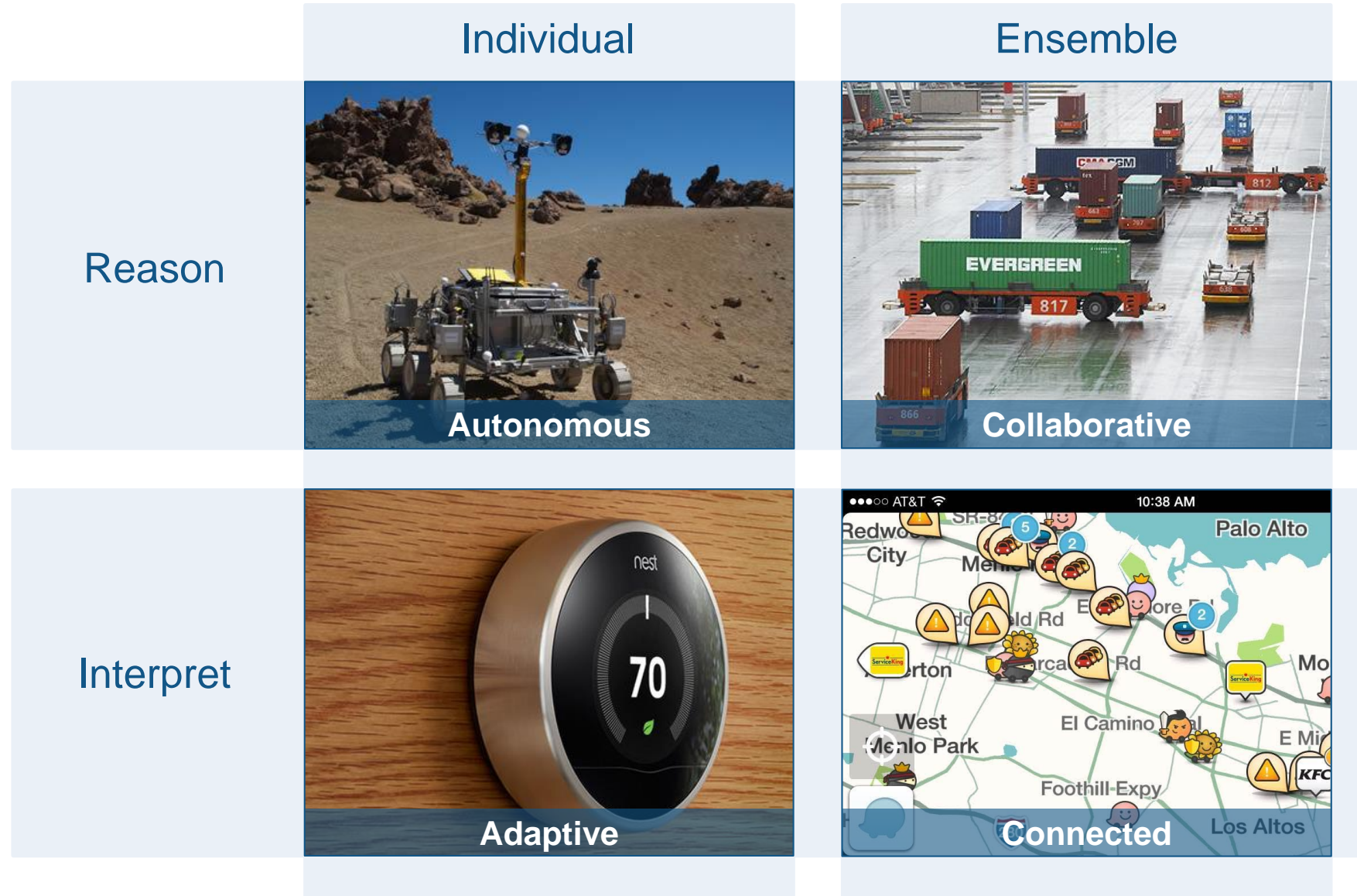




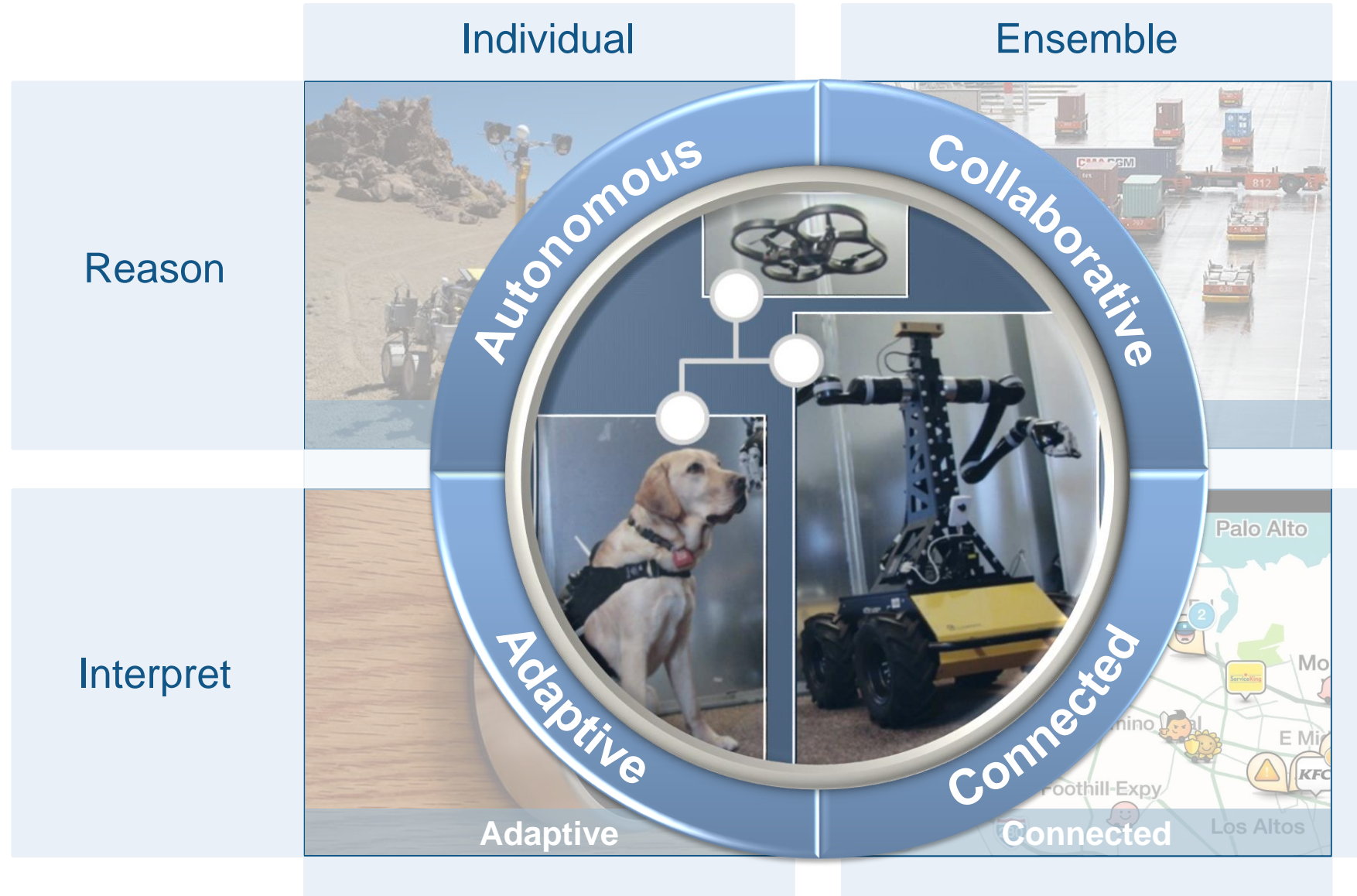
For fully-automated container handling in large terminals and terminal environments, Terex Port Solutions supplies solutions with outstanding performance.

<https://youtu.be/STlt48wXsyY?t=17s>

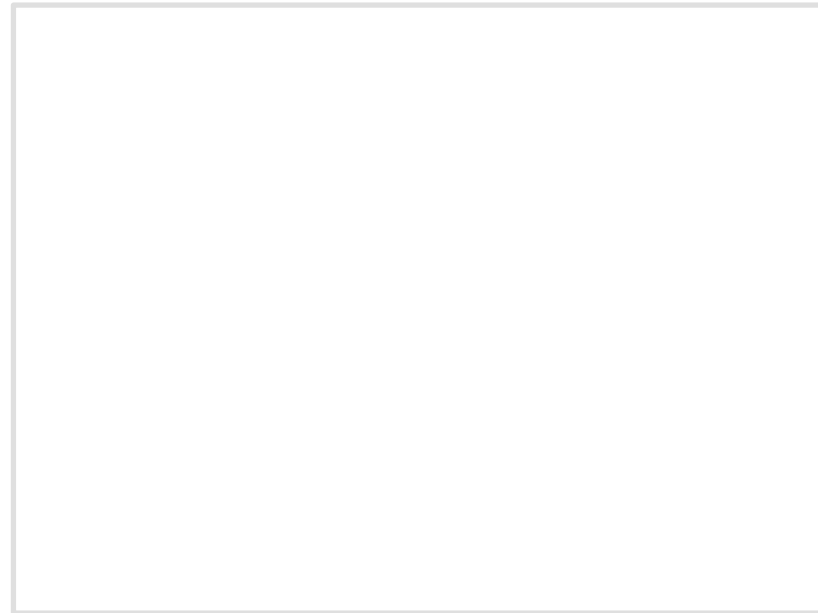
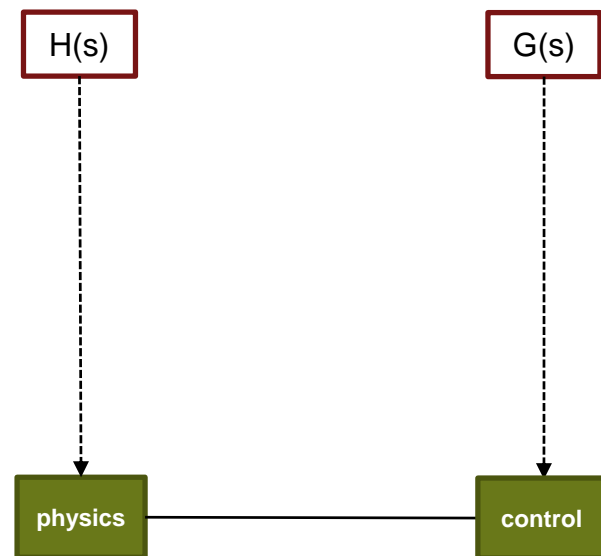
A feature classification



A feature classification

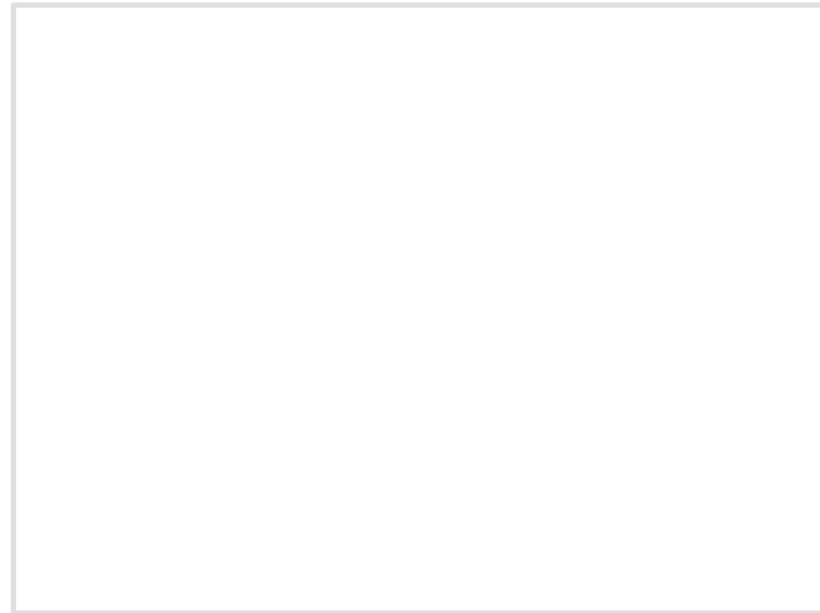
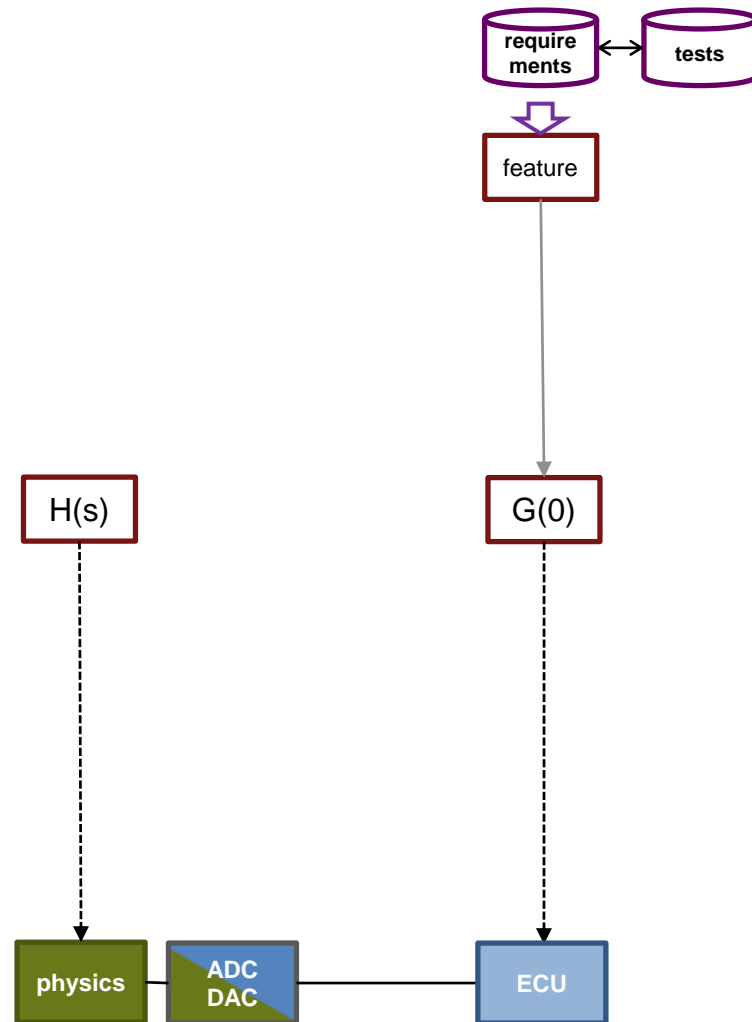




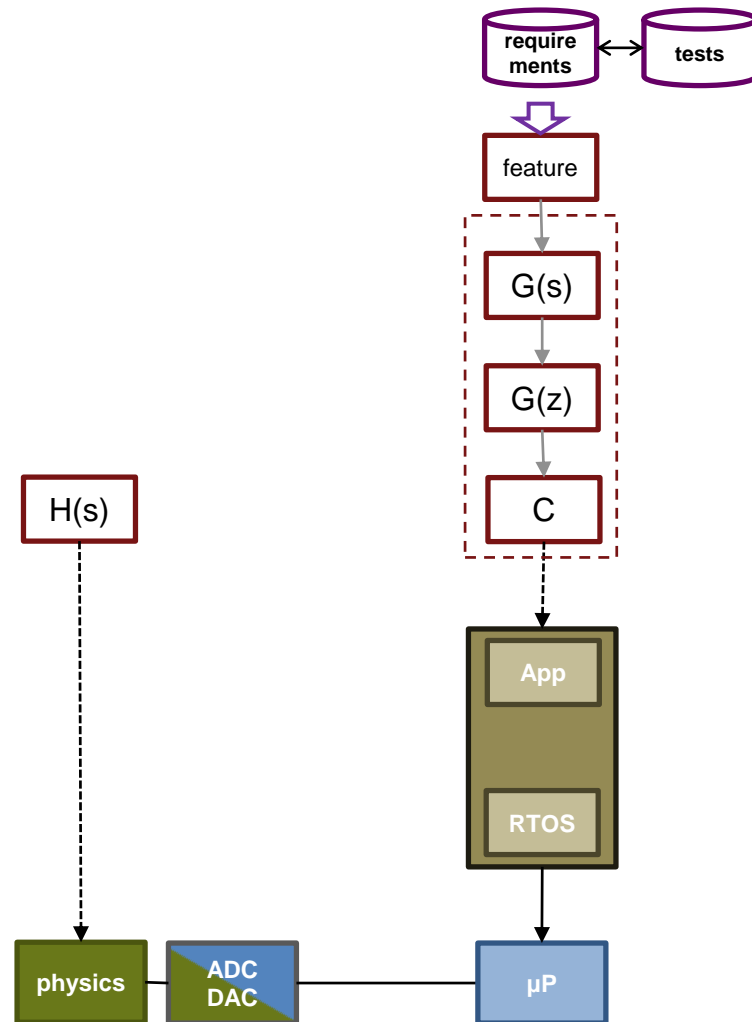


Centrifugal governor

(James Watt designed his first governor in 1788 following a suggestion from his business partner Matthew Boulton)

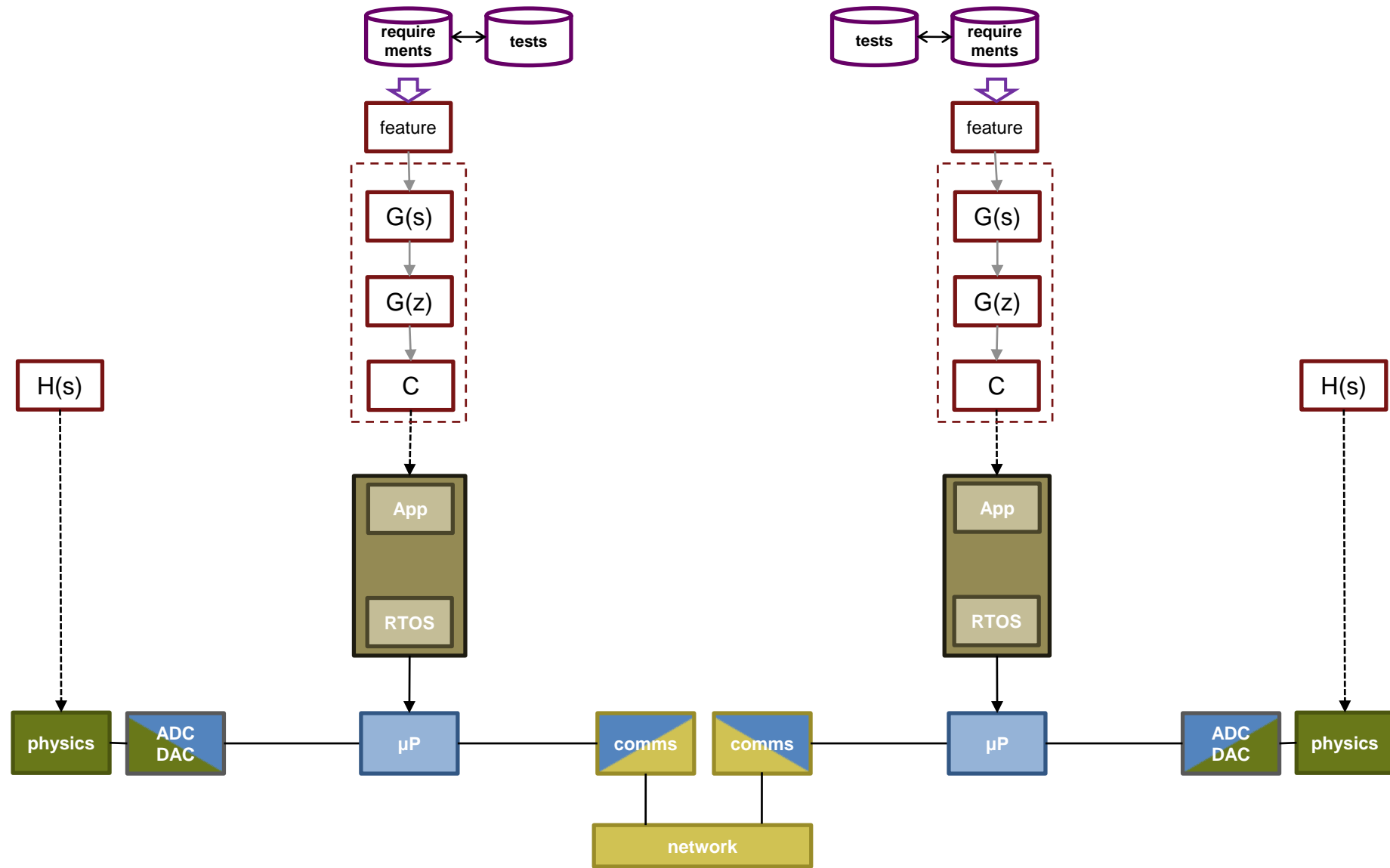


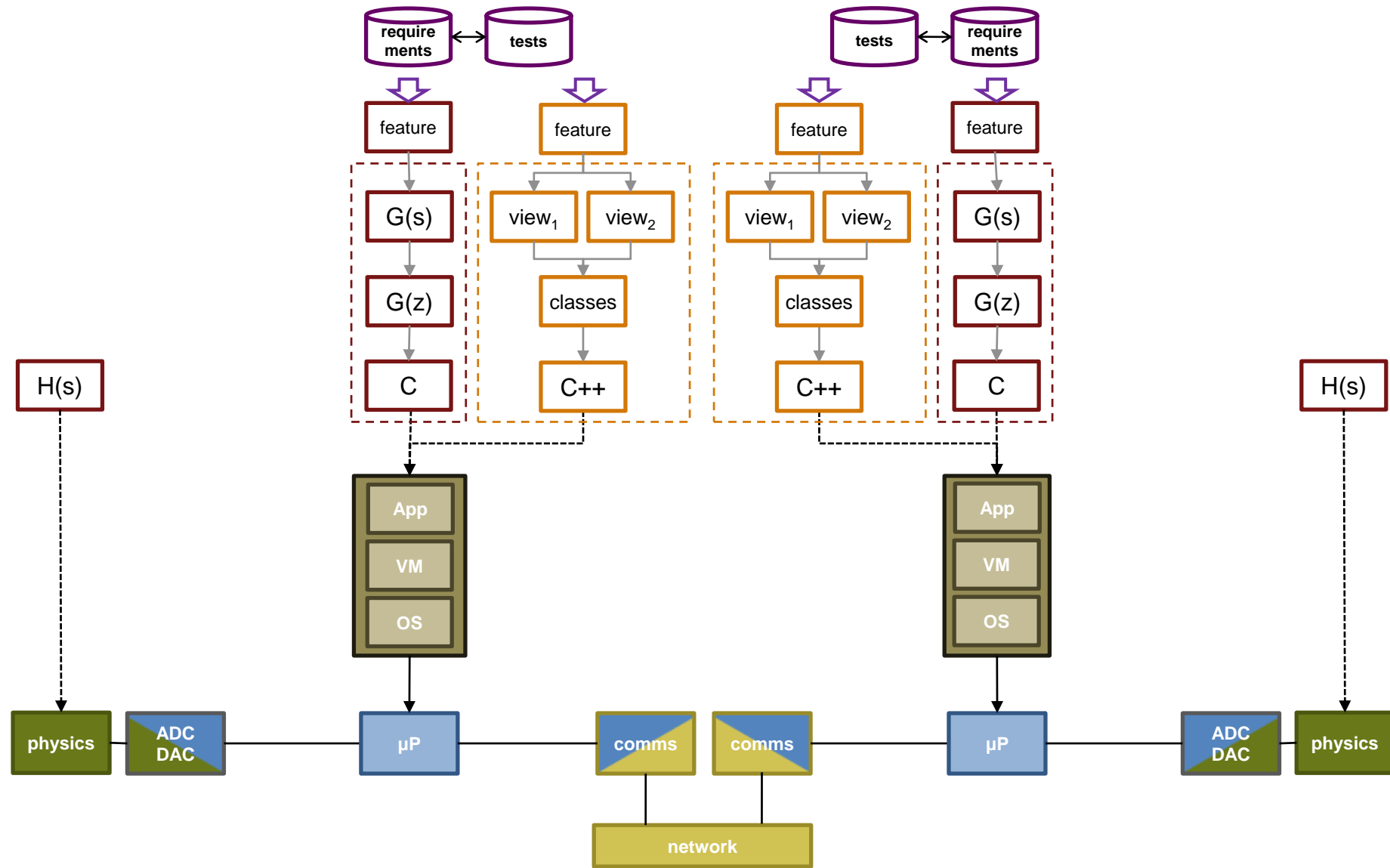
Engine control unit
(from a 1996 Chevrolet Beretta)

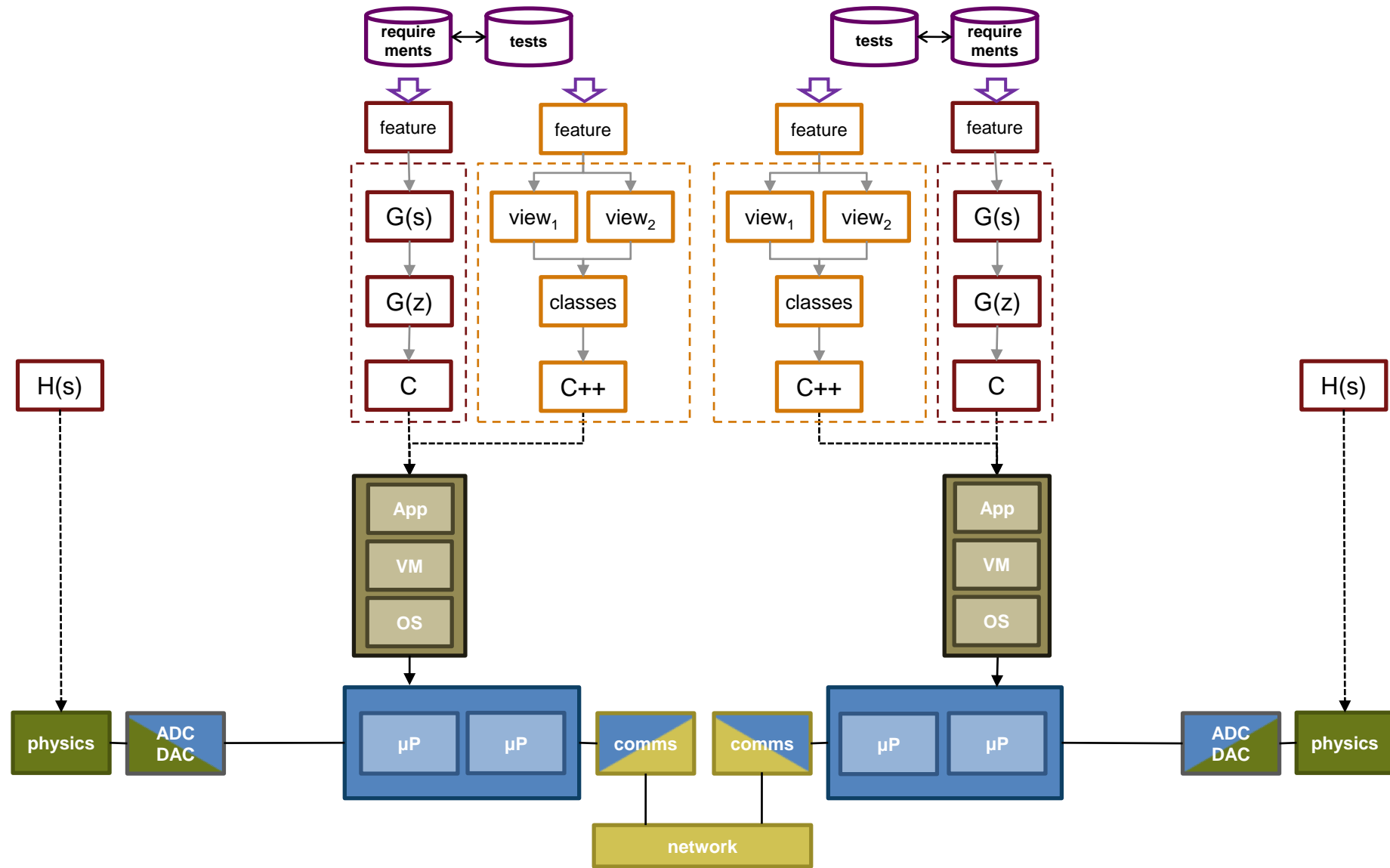


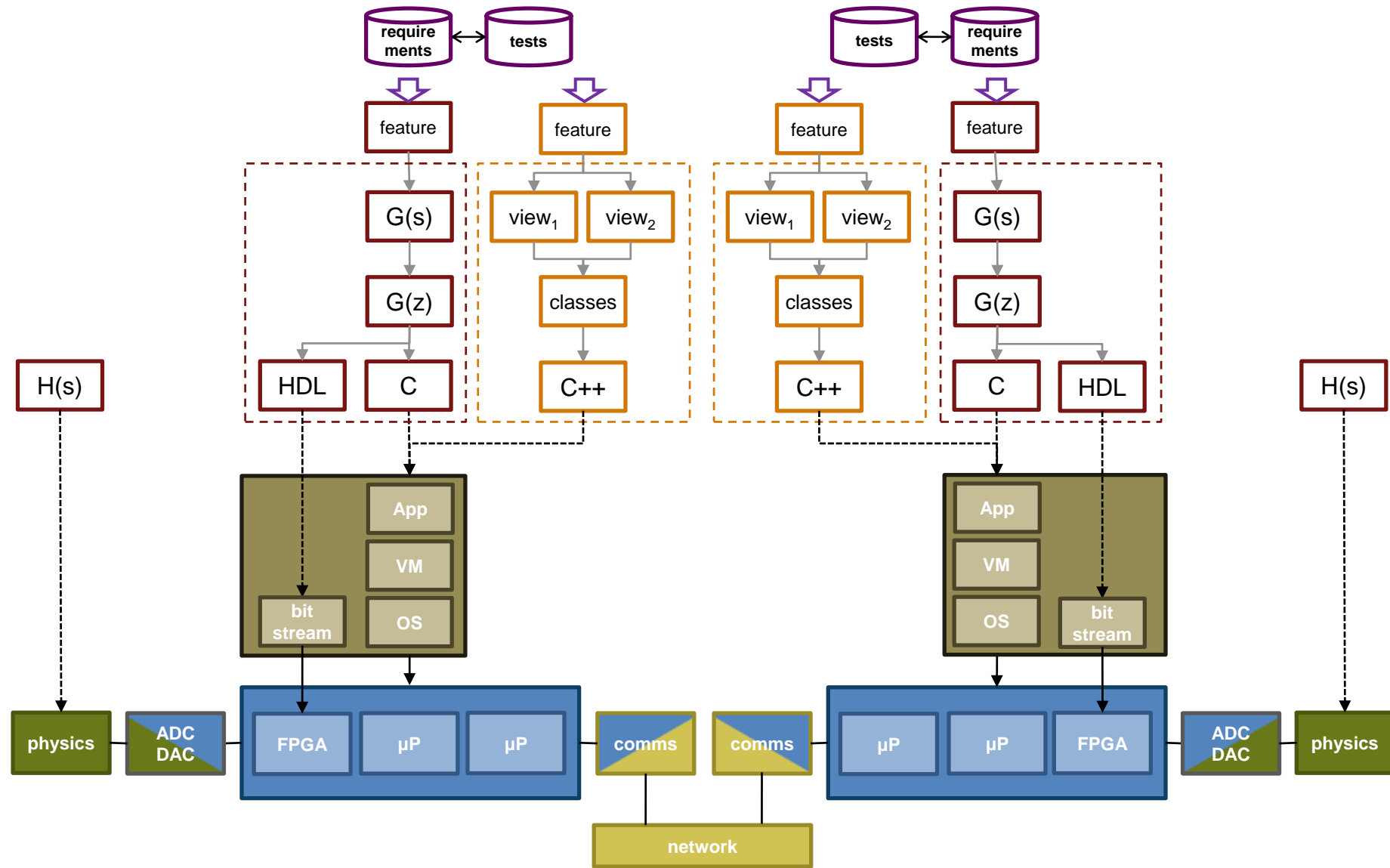
Freescle MPC561 MCU

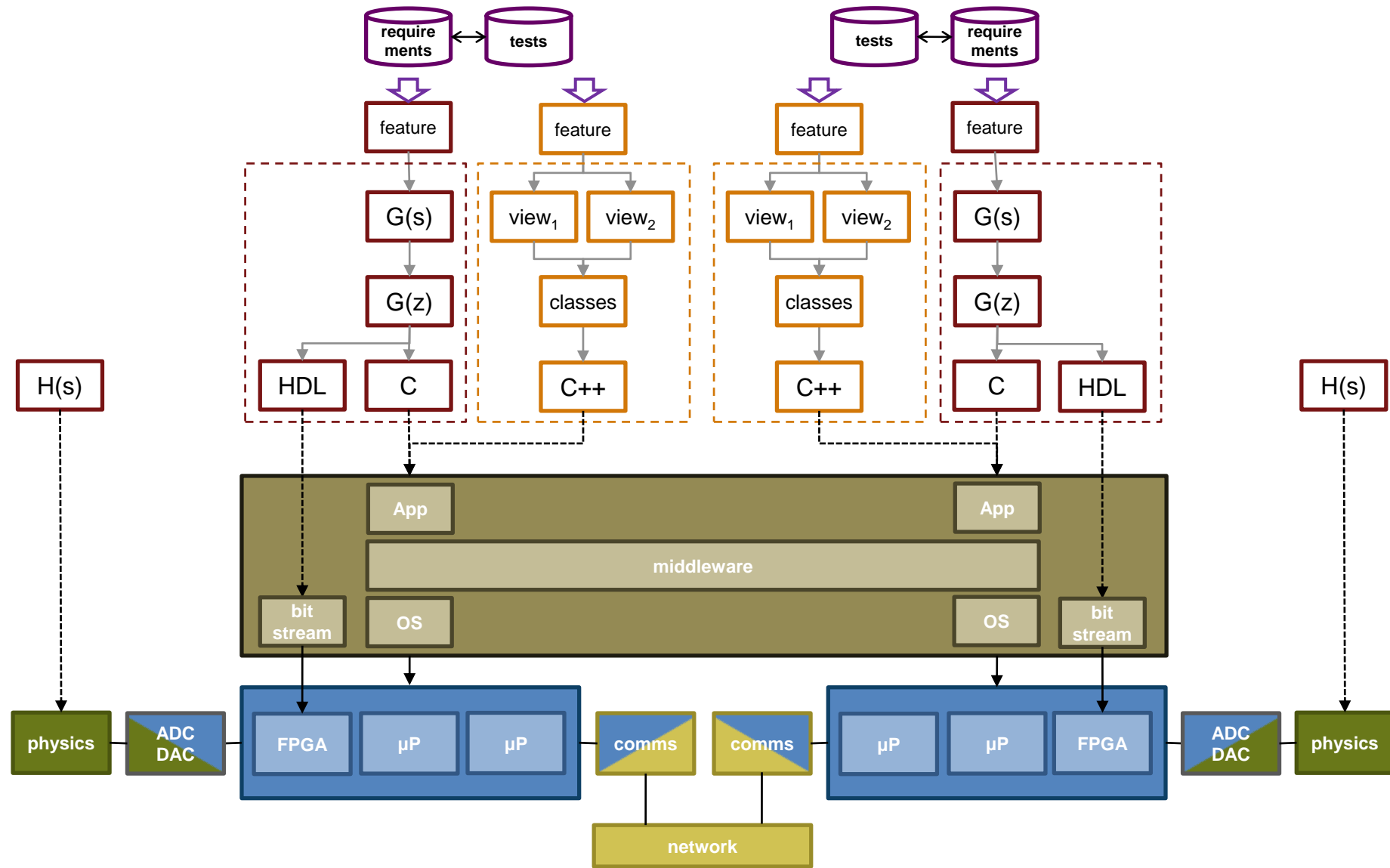
(32-bit PowerPC embedded microprocessors that operate between 40 and 66 MHz, used in engine controllers for General Motors)

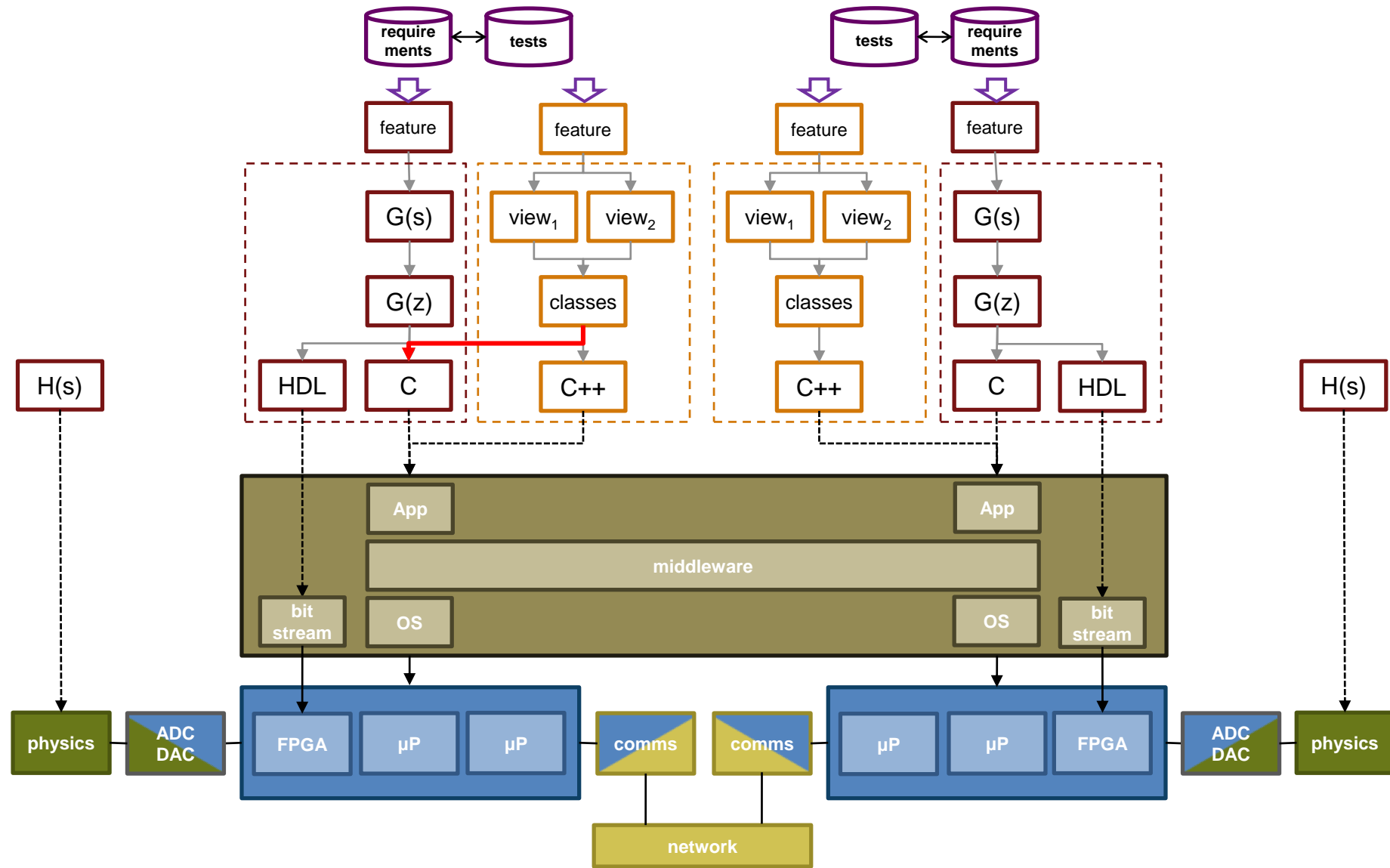


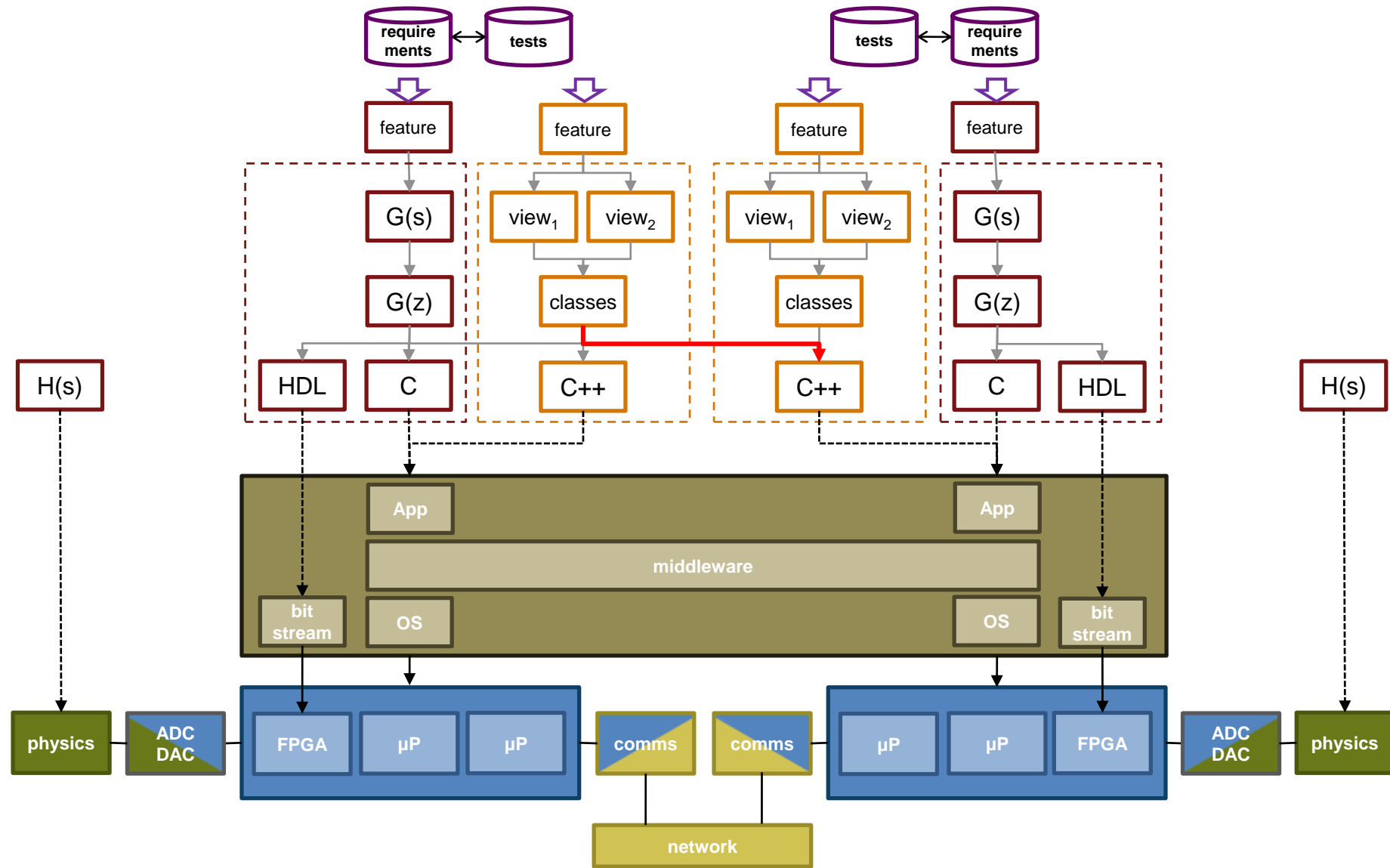


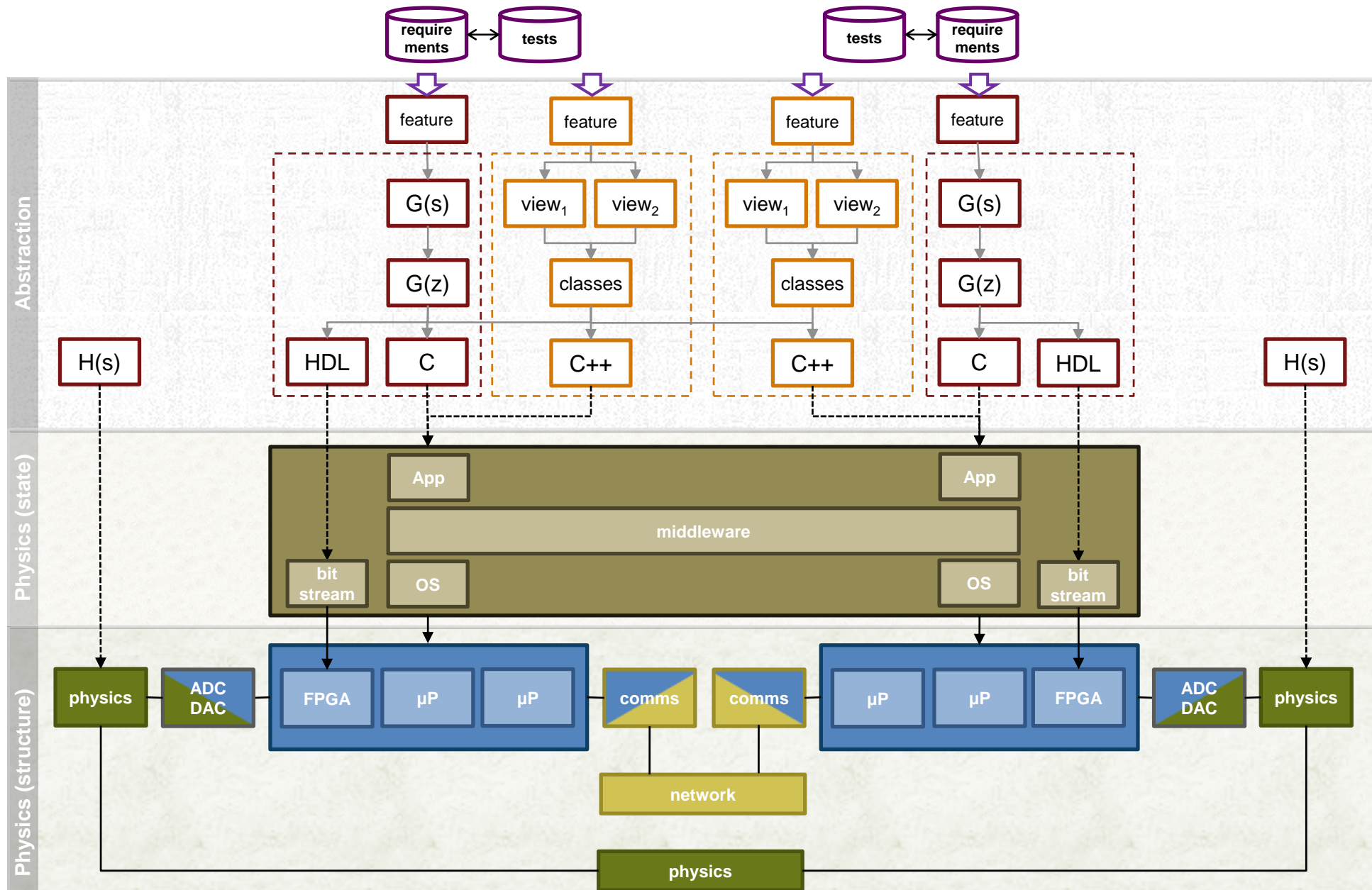


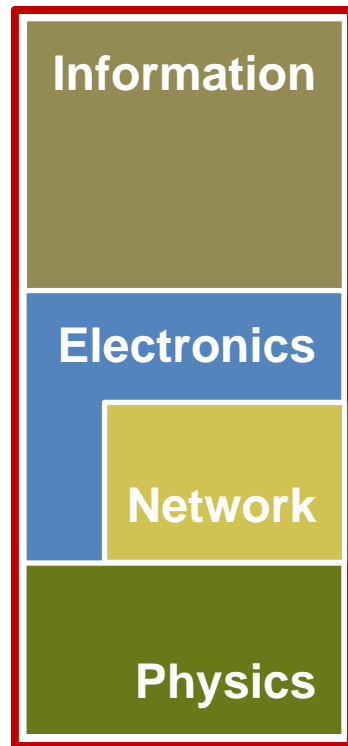


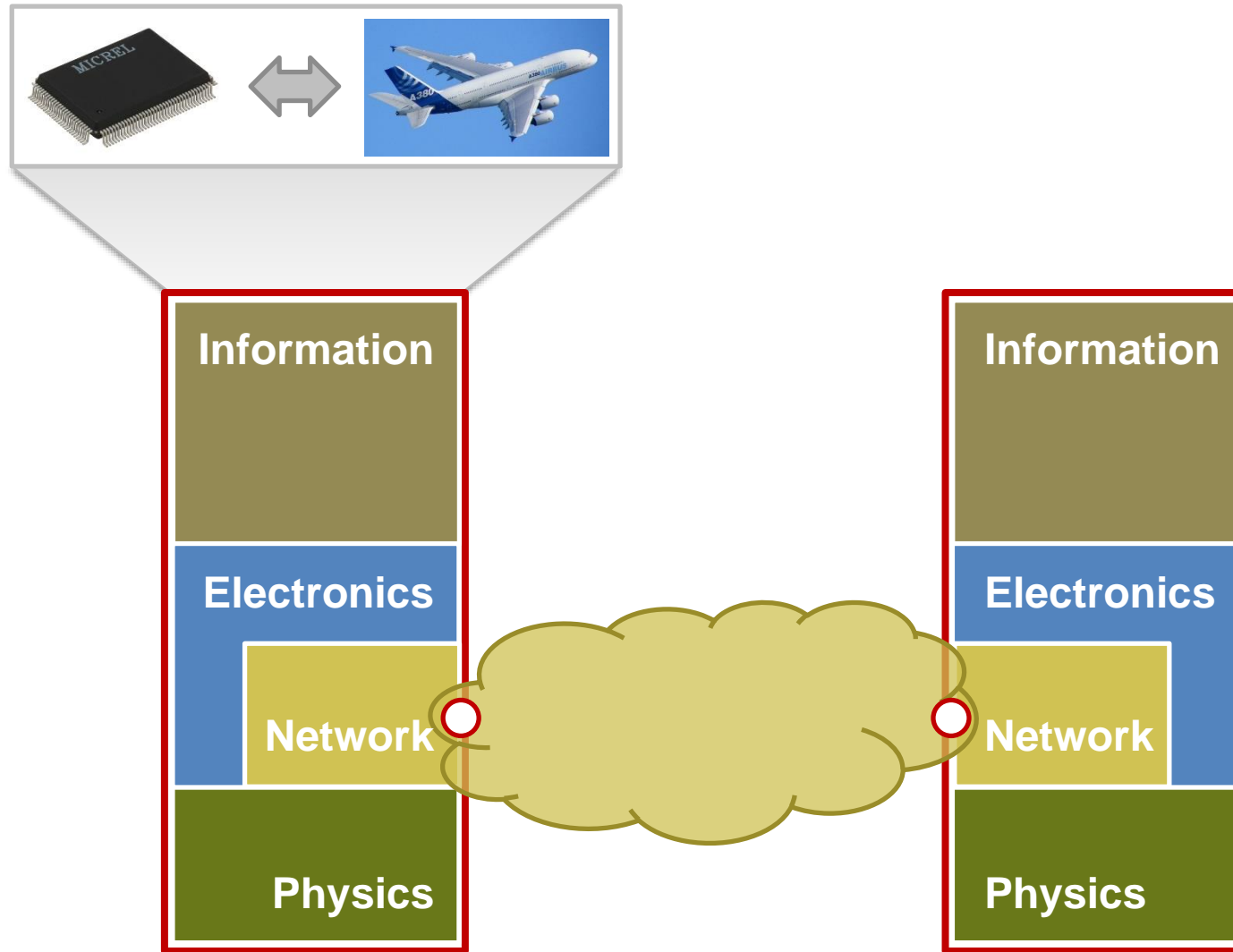


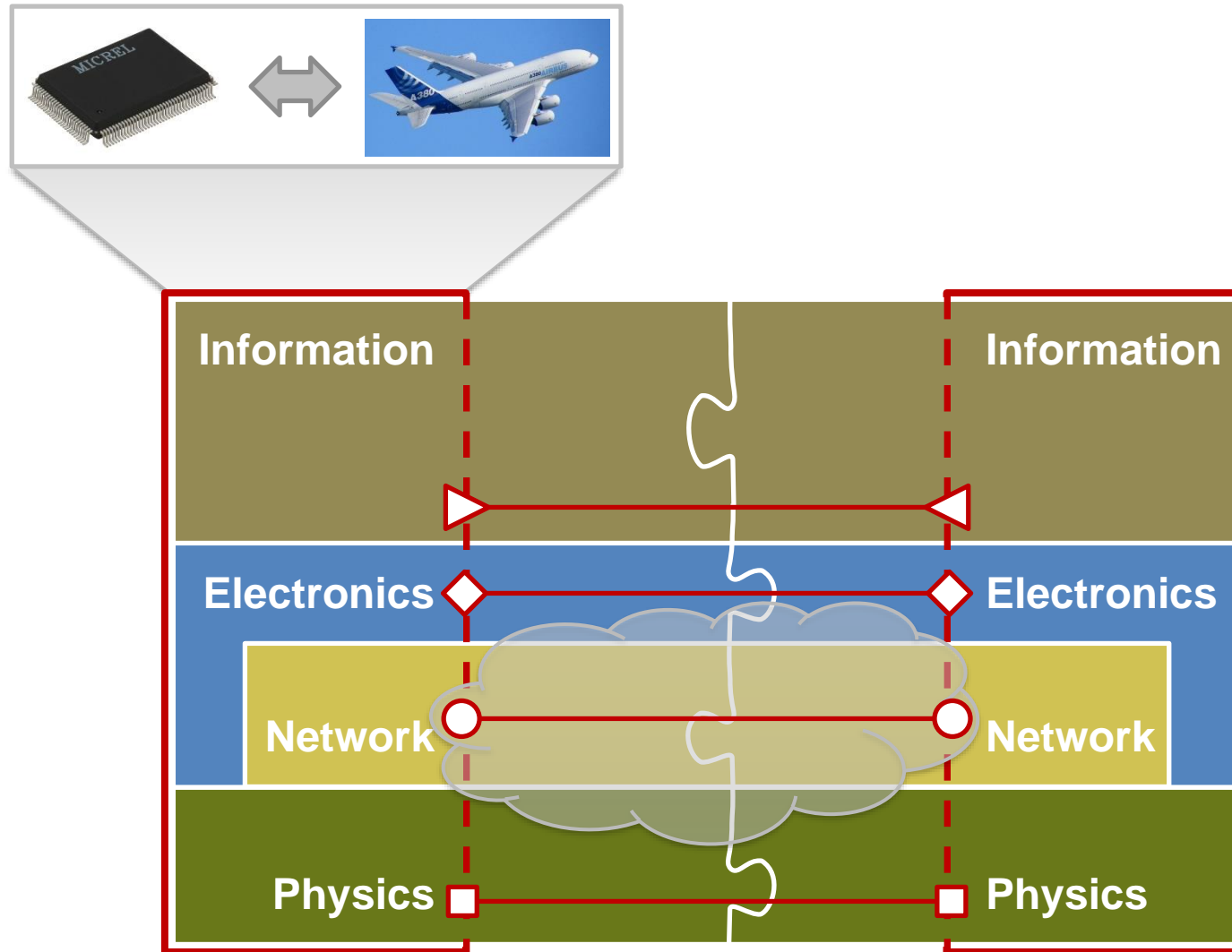


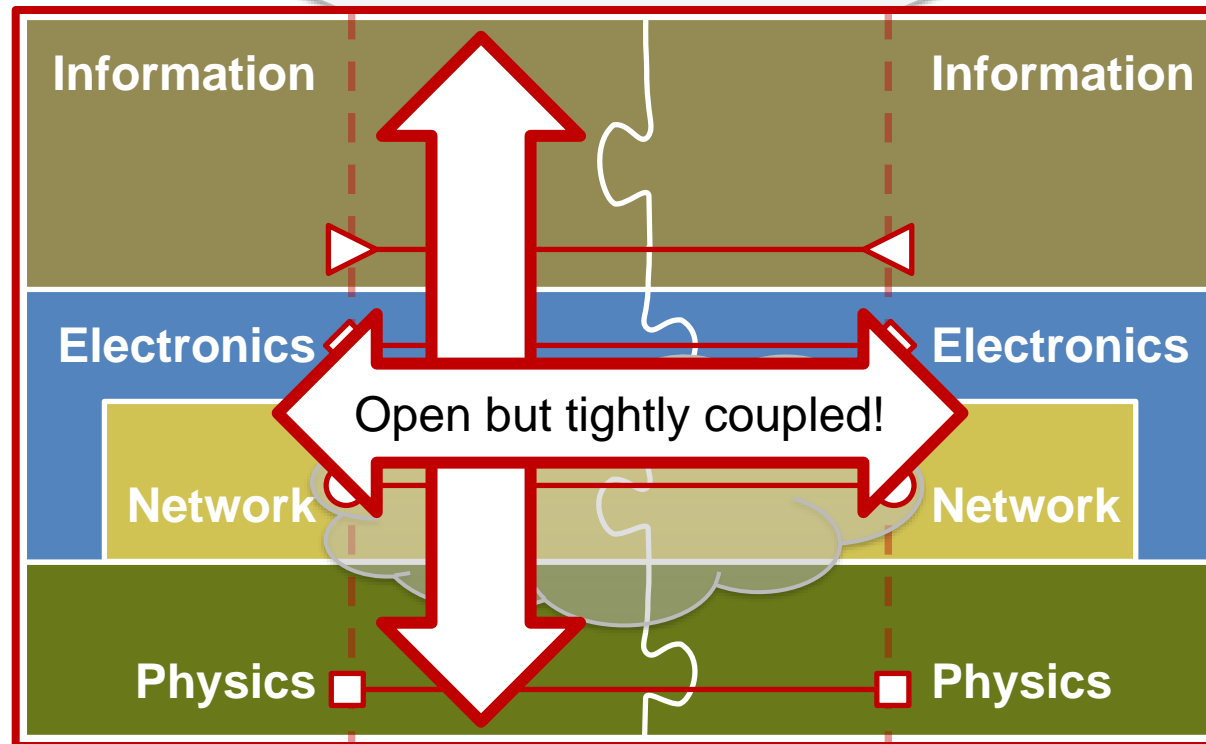
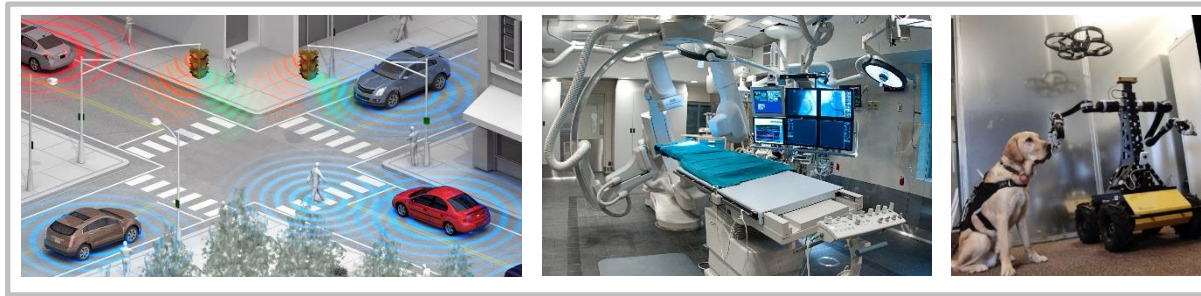














Argentinian director Fernando Livschitz of Black Sheep Films transforms a busy intersection into a choreographed dance by cloning cars, bikes, and people.

<https://youtu.be/ufK2XRGUjuc>



Connected

Exploit distributed information resources



Data sharing

Reliably configure features with varying quality of service



Wireless communication

Assemble available functionality into features after deployment



Service utilization

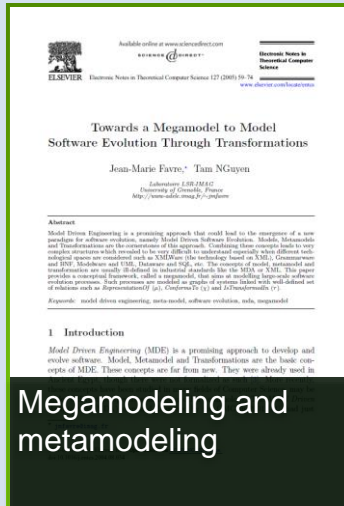
Connected

Exploit distributed information

resources



Technology challenges in CPS



Reliably configure features with varying quality of service



Wireless communication

Assemble available functionality into features after deployment



Service utilization

Data sharing

Multirate architectures

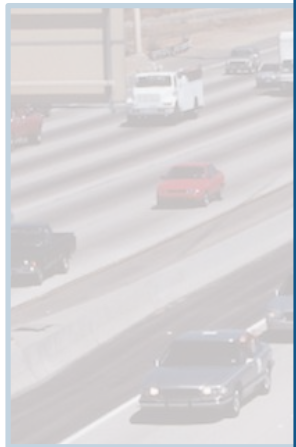
Extracting and deriving specific value from general information

Connected

Exploit distributed information resources

Reliably configure features with varying quality of service

Assemble available functionality into features after deployment



IEEE COMMUNICATIONS LETTERS / TECHNICAL NOTE, ACCEPTED FOR PUBLICATION

Standardized Protocol Stack for the Internet of (Important) Things

Maria Rita Palomaki, Member, IEEE, Nivya Achorita, Xavier Vilagras, Thomas Watanabe, Member, IEEE, Luigi Alfredo Grieco, Senior Member, IEEE, Gennaro Boggia, Senior Member, IEEE, and Michela Dabbie, Senior Member, IEEE

Abstract—The new envisioned Internet of Things (IoT) paradigm is based on the integration of heterogeneous devices (e.g., sensors, actuators, and gateways) into a single network. This paper presents a standardized protocol stack for the Internet of (Important) Things (Io(Important)T). The proposed stack is designed to support a wide range of applications, from simple sensor networks to complex industrial systems. The stack is organized into four layers: Physical, Network, Transport, and Application. The Physical layer is responsible for the transmission of data over the air interface. The Network layer is responsible for the routing of data packets. The Transport layer is responsible for the reliable transmission of data. The Application layer is responsible for the specific applications. The proposed stack is designed to be flexible and scalable, allowing for the integration of new devices and applications as needed.

Index Terms—Internet of Things (IoT), Protocol Stack, Standardization, Network, Transport, Application.

IEEE 802.15.4e low cost communication

IEEE TRANSACTIONS ON DISTRIBUTION AND MANAGEMENT, VOL. 10, NO. 4, OCTOBER 2007

An Implementation of IEEE 1588 Over IEEE 802.11b for Synchronization of Wireless Local Area Network Nodes

Tadeu Coelho, Senior Member, IEEE, John C. Eshon, Edson F. Filho, IEEE, and Mikhael Padman, Member, IEEE

Abstract—IEEE 1588 is a new standard in synchronous distributed clock timing in a network of distributed nodes. It is based on the Network Time Protocol (NTP) and is designed to provide a high level of accuracy and precision. This paper presents an implementation of IEEE 1588 over IEEE 802.11b for synchronization of wireless local area network (WLAN) nodes. The implementation is based on the IEEE 1588 standard and the IEEE 802.11b standard. The implementation is designed to be flexible and scalable, allowing for the integration of new devices and applications as needed.

Index Terms—IEEE 1588, IEEE 802.11b, Synchronization, Wireless Local Area Network (WLAN), Network Time Protocol (NTP).

IEEE 1588 precise timing

Configurable Middleware for Distributed Real-Time Systems with Aperiodic and Periodic Tasks

Xiaoming Zhang, Christopher D. Gill, Member, IEEE, Chongyan Lu, Member, IEEE

Abstract—Distributed real-time systems (DRTS) are becoming increasingly important in many applications, such as manufacturing, healthcare, and transportation. This paper presents a configurable middleware for DRTS. The middleware is designed to be flexible and scalable, allowing for the integration of new devices and applications as needed. The middleware is based on the Real-Time Executive (RTE) and the Real-Time Kernel (RTK). The RTE is responsible for the scheduling of tasks, and the RTK is responsible for the execution of tasks. The middleware is designed to be flexible and scalable, allowing for the integration of new devices and applications as needed.

Index Terms—Middleware, Distributed Real-Time Systems, Aperiodic and Periodic Tasks, Real-Time Executive (RTE), Real-Time Kernel (RTK).

Distributed real-time systems task scheduling

A Survey on Standards for Real-Time Distribution Middleware

HÉCTOR PÉREZ and J. JAVIER GUTIÉRREZ, University of Córdoba

Abstract—The development of distributed real-time systems (DRTS) is a complex task that requires the integration of many different technologies. This paper presents a survey on standards for real-time distribution middleware. The survey is based on the Real-Time Executive (RTE) and the Real-Time Kernel (RTK). The RTE is responsible for the scheduling of tasks, and the RTK is responsible for the execution of tasks. The survey is designed to be flexible and scalable, allowing for the integration of new devices and applications as needed.

Index Terms—Standards, Real-Time Distribution Middleware, Real-Time Executive (RTE), Real-Time Kernel (RTK).

Processor and network scheduling



Data sharing

Wireless communication

Service utilization

Multirate architectures

Extracting and deriving specific value from general information

Physically aware configurable protocol stack that is IP compatible

Precise timing and synchronization in a distributed environment

Data sharing

Multirate architectures

Extracting and deriving specific value from general information

Reliably configure features with varying quality of service

Wireless communication

Physically aware configurable
protocol stack that is IP
compatible

Precise timing and synchronization in a distributed environment

Assemble available functionality into features after deployment

[illegible]

Service utilization

Real-time embedded services operating in a physical environment

Smart services discovery

Information sharing in a heterogeneous system ensemble

Collaborative

Reliably configure an ensemble online to exploit exogenous functionality



Runtime system adaptation

Contract out endogenous resources and balance use of exogenous resources



Hardware resource sharing

Purpose functionality to create novel system features post deployment



Functionality sharing

Collaborative

Reliably configure an ensemble
online to exploit exogenous

Mechanisms for Leveraging Models at Runtime in Self-adaptive Software

Amel Benacerruf¹, Robert Figueira², Giacomo Tamburini³, Thomas Vignati⁴,
Pierre J. Montemanni⁵, Walter Cusack⁶, Fabio M. Costa⁷,
Albano Pierantonio⁸, Matthias Telsi⁹, Mehmet Akat¹⁰, Pia Emmannsson¹¹,
Hong Guo¹², Nicholas Georgantas¹³, and David Briffles¹⁴

¹ Inria, France
² Colorado State University, US
³ Università della Svizzera Italiana, Switzerland
⁴ Hasso Plattner Institute at the University of Potsdam, Germany
⁵ MathWorks, US
⁶ Università degli Studi di Milano, Italy
⁷ Universidade Federal do Rio de Janeiro, Brazil
⁸ Univ. degli Studi di L'Aquila, Italy
⁹ Chalmers, University of Gothenburg, Sweden
¹⁰ University of Twente, Netherlands
¹¹ Erasmus AI, Sweden
¹² Peking University, China
¹³ Lancaster University, UK

Abstract. Modern software systems are often required to adapt their behavior at runtime to maintain or enhance their utility in dynamic environments. Models at runtime research aims to provide suitable abstractions, techniques, and tools to manage the complexity of adapting software systems at runtime. In this chapter, we discuss challenges associated with developing mechanisms that leverage models at runtime to support runtime software adaptation. Specifically, we discuss challenges associated with developing effective mechanisms for supporting runtime systems, reasoning about and planning adaptations, maintaining consistency among multiple runtime models, and maintaining fidelity of runtime models with respect to the running system and its environment. We discuss related problems and state-of-the-art mechanisms, and identify open research challenges.

Models @ runtime

Available online at www.sciencedirect.com
ScienceDirect
Procedia Computer Science 161 (2019) 1–7

International Conference on Computational Science, ICCS 2019

Towards automated model calibration and validation in rail transit simulation

Yilin Huang, Manabro D. Seck, Alexander Vohwinkel
System Engineering Group, Faculty of Technology, Policy and Management
Imperial College of Science, Technology and Innovation

Abstract

The benefits of modeling and simulation in rail transit operations has been demonstrated in various studies. However, the complex dynamics involved and the ever-changing environment in which rail systems evolve expose the limits of classical simulation. Changing environmental conditions and second order dynamics challenge the validity of the models and seriously reduce model usability. This paper discusses the potential benefits and requirements of dynamic data driven simulation in rail systems. The emphasis is placed on automated model reconfiguration, calibration, and validation through the use of data analysis methods. The rationale and requirements are discussed and a process model for data driven calibration and validation is proposed.

Keywords: Public rail transport simulation, model validation, calibration.

1. Introduction

Rail transport systems have inherent long life spans. In addition to their well-known dynamic characteristics known as "first order dynamics", the system themselves are also subject to change, which is referred to as their "second order dynamics" [1]. In public rail transport systems, daily/weekly/seasonal patterns of vehicle ridership, and the distribution profile of vehicles are examples of first-order dynamics, second order dynamics often denote

Automated model
calibration

Contract out endogenous
resources and balance use of
exogenous resources



Hardware resource sharing

Purpose functionality to create
novel system features post
deployment



Functionality sharing

Runtime system adaptation

Reasoning and planning
adaptation of an ensemble of
systems

Purpose functionality to create novel system features post deployment



Functionality sharing

Flexible and transferable
embedded functionality dispatch

Performance characterization
from abstract functionality

Contract out endogenous resources and balance use of

Purpose functionality to create novel system features post



Addressing the Challenges of Tactical Information Management in Net-Centric Systems with DDS

David C. Schmidt (Google Inc., Seattle, WA)
PamelaK Corporation,
4 Lincoln Park Lane, Suite 100, Harrison, NJ, 07033, USA
E-mail: dschid@pamela.com; P.K.Corp; P.K.Corp LLC
Jong-chul Jung (Korea IT-2177 Project) 103-20-100, Seoul
Jong-chul.jung@it2177project.com

Abstract

Recent trends in net-centric systems, including the distributed nature of tactical information management capabilities that require the right information at the right place at the right time to satisfy quality of service (QoS) requirements in heterogeneous environments. This article presents an architectural overview of the Object Management Group (OMG)'s Data Distribution Service (DDS) standard, which defines a set of interfaces for creating middleware platforms that enable applications to dynamically discover and use information resources. The use and deployment of information they need in a timely manner. DDS is an important distributed software technology for mission-critical real-time control systems because it supports 100+ location independent, non-affiliated nodes. It implements the processes that enable communication between distributed nodes and manages published information. It is designed to support large numbers of topics, and to support a wide range of QoS requirements, including information use standard interfaces and transport protocols.

1 Introduction

Tactical information management systems increasingly use net-centric architectures characterized by thousands of platforms, sensors, devices and nodes, and components interacting to exchange information. These systems are subject to a wide range of dynamic changes, and often changes in the physical environment. For example, the number of nodes in the network can vary, and some nodes may be destroyed. These systems also combine new information processing being designed to move from different sensors and processing nodes, as well as individuals participating in specific missions, can collaborate effectively and deliver information in a timely, predictable, and secure manner.

Data distribution

[illegible]

Multi-rate double

Functionality sharing

Multi-use functionality post-deployment

Feature interaction

Design

Confidently design systems as part of a reliable ensemble



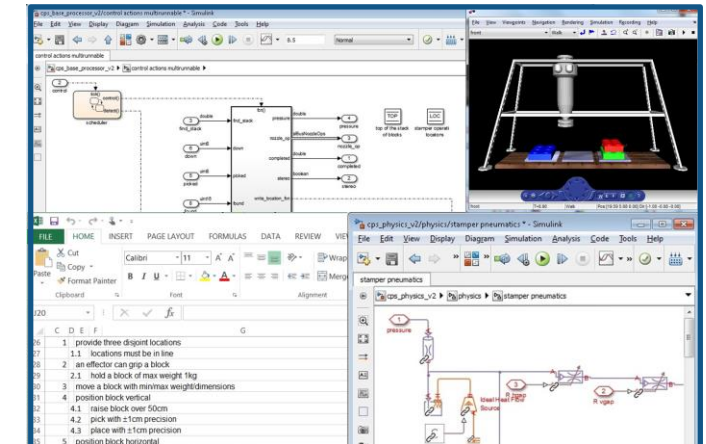
Virtual system integration

Systematically design optimal behavior of system ensembles



Emerging behavior design

Collaborate between stakeholders throughout the system life cycle



Design artifact sharing

Design

Confidently design systems as part of a reliable ensemble

Systematically design optimal behavior of system ensembles

Collaborate between stakeholders throughout the

Model Building Automation System

Counterexample guided abstraction refinement

Multiparadigm modeling

Hybrid dynamic systems

Hyperdense time domain for hybrid bond graphs

Real-time simulation

Virtual system integration

Emerging behavior design

Design artifact sharing

Proper models in design

System-level design and analysis by using models

Connectivity among models, software, and hardware

Design

Confidently design systems as part of a reliable ensemble



Virtual system integration

Proper models in design

System-level design and analysis by using models

Connectivity among models, software, and hardware

Systematically design optimal behavior of system ensembles

Technical Report MSR-TR-2005-28, February, 2005

Towards Service-Oriented Networked Embedded Computing

Jie Lin and Feng Zhao
Microsoft Research
One Microsoft Way
Redmond, WA 98052, USA
{jliu, fzhao}@microsoft.com

Liz is a site manager at the high-rise Citicorp office building in downtown B city. As one of their major customers has just moved to another site, she wants to evaluate the idea of opening part of the parking space underneath the building to the general public. To help her to reach a decision, she wants to collect vehicle arrival and departure statistics in the parking lot for a period of two weeks. Pablo is a security officer for the Citicorp building. He is investigating complaints from people that there are several cars driving extremely fast in certain areas of the garage. He wants to take pictures of these cars and issue warnings to offending drivers. Camer is a local law enforcement agent in B city. He is in charge of handling chemical sensors at strategic locations throughout the city for terrorism detection. He has installed some of them in the Citicorp building garage, and wants a notification whenever a vehicle carrying certain chemical elements is detected. Although they are from different organizations, Liz, Pablo, and Camer all plan to use a generic wireless sensor network recently installed in the garage and augment it with special purpose sensors as needed.

Systems like these are not readily supported by the current networked embedded system

Service oriented sensor programming

data can be either aggregated through routing or processed centrally. Although all sensors have been made on running user queries within the network, the architecture is ill-suited for monitoring and tracking specific events in a resource-constrained environment. The application-specific view (Shore) acknowledges the fact that many sensor network system behaviors depend heavily on the physical stimuli, and tries to make the best use of the power and bandwidth resources through exploiting the application-specific dynamics. For example, a microphone-based vehicle tracking system would be designed quite differently from a camera-based system, although many of the system components are similar. Everything is optimized at the design time. The system is rigid and hard to change afterwards.

These two philosophies can be viewed as two extremes for handling application logics. In the data collection view, there is no application logic in the network; everything is processed off line. It is quite generic but not always resource optimal. In the application specific view, the application logic is hard wired into the network. The designers have fine grained resource control but the system is rigid and hard to reuse. We motivate here a service-oriented architecture for networked

Machine ballets don't need conductors

Towards scheduling-based service choreographies in a real-time SOA for industrial automation

Thomas Kollmann, Alfons Kemper
Chair for Database Systems
Technische Universität München, Germany
{tollmann, kemper}@db.in.tum.de



Abstract—Today's manufacturing industry is under pressure to increase the flexibility of its factory floor. One approach to achieve this goal is the shift from centralized control systems towards distributed, service-oriented architectures (SOAs). To fully leverage the benefits of this new paradigm, the SOA should extend down to the device and not only virtual resource-constrained devices, such as smart sensors and actuators. In this paper we present our approach for a lightweight distributed service choreography without a central point of control. It is based on network-aware representations of a state, non-deterministic scheduling, and a distributed set of stateful, non-deterministic devices. In contrast to previous work, our focus lies on the planning components required for achieving a service choreography. These scheduling is a central part of our architecture, and we report on its in-situ execution from within the planning process. We envision different business use cases.

1. INTRODUCTION

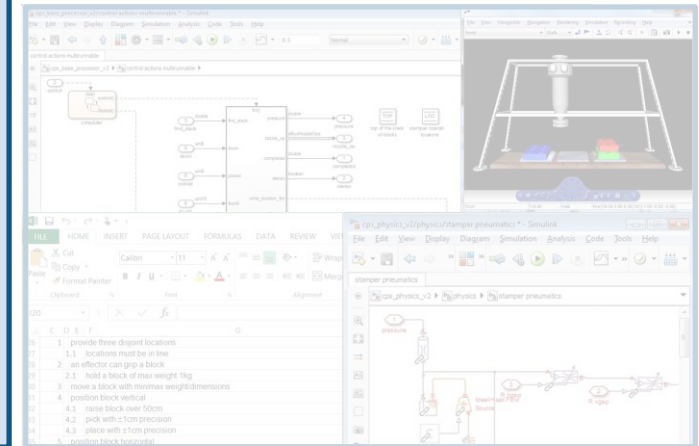
Manufacturing and process industries are under pressure from external product life-cycles, market fit and demands for reducing the time to market for each new product. Traditional manufacturing plants do not well meet these new requirements because the time for installation and setup are continuing a considerable portion of the total life-cycle costs [1]. The problem stems from the inflexible design of traditional manufacturing systems. Centralized control and scan-based control systems are optimized for a given physical and network configuration. They are tightly coupled with their environment which hinders reconfiguration and reuse. The hardware side is already changing slowly, industrial Ethernet is gaining traction [2] while sensor embedded devices are performing an increasing number of orthogonal tasks, for example wireless sensor networks (WSNs) monitoring machine health. These developments call for the presence in the

Service orchestration

Emerging behavior design

Collaborative planning, guidance, and control

Collaborate between stakeholders throughout the system life cycle



Design artifact sharing

Design

Confidently design systems as part of a reliable ensemble



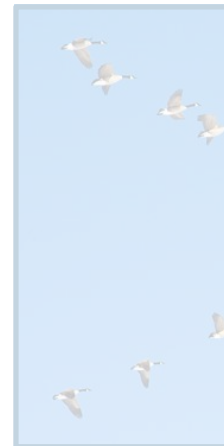
Virtual system integration

Proper models in design

System-level design and analysis by using models

Connectivity among models, software, and hardware

Systematically design optimal behavior of system ensembles



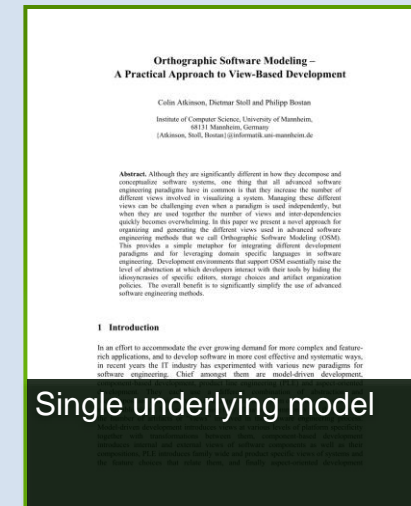
Emerging behavior design

Collaborative planning, guidance, and control

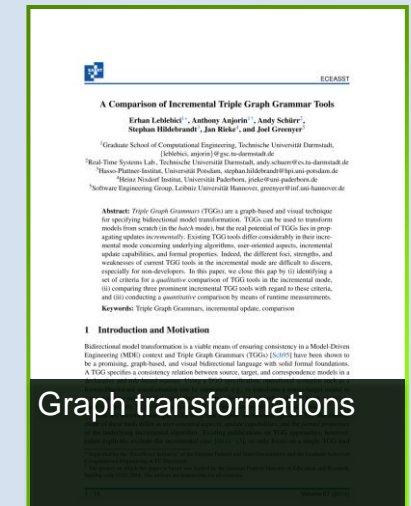
Collaborate between stakeholders throughout the design process



Open services for lifecycle collaboration



Single underlying model



Graph transformations

Design artifact sharing

Tool coupling among disparate organizations

Support manifold views and tools in design

Softw Syst Model
DOI 10.1007/s10270-015-0469-x

INDUSTRY VOICE

Cyber-physical systems challenges: a needs analysis for collaborating embedded software systems

Pieter J. Mosterman¹ · Justyna Zander²

© Springer-Verlag Berlin Heidelberg 2015

Abstract Embedding computing power in a physical environment has provided the functional flexibility and performance necessary in modern products such as automobiles, aircraft, smartphones, and more. As product features came to increasingly rely on software, a network infrastructure helped factor out common hardware and offered sharing functionality for further innovation. A logical consequence was the need for system integration. Even in the case of a single original end manufacturer who is responsible for the final product, system integration is quite a challenge. More recently, there have been systems coming online that must perform system integration even after deployment—that is, during operation. This has given rise to the cyber-physical systems (CPS) paradigm. In this paper, select key enablers for a new type of system integration are discussed. The needs and challenges for designing and operating CPS are identified along with corresponding technologies to address the challenges and their potential impact. The intent is to contribute to a model-based research agenda in terms of design methods, implementation technologies, and organization challenges necessary to bring the next-generation systems online.

Keywords Cyber-physical systems · Computation · Embedded systems · Challenges · Internet of Things · Modeling and simulation

1 Motivation

Engineered systems rely on ingenuity and technology to implement a desired functionality, examples of which include aircraft, automobiles, power plants, smartphones, robots, washers and dryers, pacemakers, and more. Embedded systems are engineered systems that implement functionality by employing computational technologies. The embedded nature allows the computational elements to interact directly (i) with a physical computing platform that it executes on and (ii) with its physical surroundings. In other words, computational logic may obtain input from sensors that measure physical quantities, execute physical instructions of a computing platform to compute output from this input, and provide the output to actuators that effect change in physical quantities and affect the physical behavior.

The intent of this paper is to explore the maturation of embedded systems and the evolution of the concept of cyber-physical systems (CPS). A result of this exploration is the identification of challenges specific to systems of a CPS nature. The perspective reflects upon an industry vantage point. Focus is on models for solving industry-relevant challenges when developing next-generation software systems. While the material is intended to be accessible to the

Communicated by Tony Clark, Gabor Karsai, and Roel J. Wieringa.

✉ Justyna Zander
justyna.zander@gmail.com; dr.justyna.zander@ieee.org
Pieter J. Mosterman

Pieter J. Mosterman and Justyna Zander, “Cyber-physical systems challenges: a needs analysis for collaborating embedded software systems,” in *Software & Systems Modeling*, Springer Berlin/Heidelberg, ISSN 1619-1366, vol. 15, nr. 1, pp. 5-16, 2016



Conclusions

- World is becoming machines that
 - Adapt to the environment
 - Make decisions autonomously
 - Are connected
 - Work together
- Metcalfe is supplanting Moore as value driver
- Key M&S application areas
 - Performance
 - Concurrency
 - Physics
- We need a stupendous range of technologies combined
 - Do not be an individualist!
- MathWorks touches on most any of these technologies
- Come seek us out to discuss!
 - Opportunities
 - Solution needs
 - Starting points



